JPRS L/10225

30 December 1981

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 26/81)

FBIS FOREIGN BROADCAST INFORMATION SERVICE

FOR OFFICIAL USE ONLY

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000500010023-0

NOTE

JPRS publications contain information primarily from foreign newspapers, periodicals and books, but also from news agency transmissions and broadcasts. Materials from foreign-language sources are translated; those from English-language sources are transcribed or reprinted, with the original phrasing and other characteristics retained.

Headlines, editorial reports, and material enclosed in brackets [] are supplied by JPRS. Processing indicators such as [Text] or [Excerpt] in the first line of each item, or following the last line of a brief, indicate how the original information was processed. Where no processing indicator is given, the information was summarized or extracted.

Unfamiliar names rendered phonetically or transliterated are enclosed in parentheses. Words or names preceded by a question mark and enclosed in parentheses were not clear in the original but have been supplied as appropriate in context. Other unattributed parenthetical notes within the body of an item originate with the source. Times within items are as given by source.

The contents of this publication in no way represent the policies, views or attitudes of the U.S. Government.

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000500010023-0

JPRS L/10225

30 December 1981

USSR REPORT

Cybernetics, Computers and Automation Technology

(FOUO 26/81)

Contents

GENERAL

SURVEY OF DEVELOPMENTS IN CYBERNETICS BY EDITORIAL BOARD

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 1-12

[Series of comments by members of editorial board]

[Text] This anniversary issue of the journal KIBERNETIKA, issue No 100, is dedicated to the 26th Congress of the CPSU, a great event in the life of our country. In the short comments below, members of the editorial board discuss plans for carrying out the tasks given to cybernetics scientists by the Congress for the 11th Five-Year Plan.

V. M. Glushkov, academician and editor-in-chief of the journal.

Ukrainian cybernetics scientists are directing all their efforts to carrying out the program of action adopted at the 26th Congress of the Communist Party of the Soviet Union. The concern for preservation of the peace obligates us to make every effort to see that peace on earth is preserved.

Workers in science face enormous tasks in the 11th Five-Year Plan: Improving computer hardware and control systems, constructing new cybernetics systems capable of efficiently aiding human beings in many spheres of activity, in controlling highly complex instruments and mechanisms, production of output, and planning and designing large systems. The most important challenges facing researchers were reflected in the materials of the 26th CPSU Congress. They are, first of all, the construction and employment of systems of different levels and purposes, in particular the OGAS (State Automated System for Data Collection and Processing), which is expected to serve as a tool for decision-making in management. The development of promising types of hardware, especially recursive computers constructed on entirely new principles for organization of computations and computer components, is one of the central challenges in building highly productive computer hardware. Other very important tasks are the building and practical introduction of robot-manipulators and other cybernetics systems with artificial intellect and setting up automated design systems for various technical objects and structures such as airplanes, atomic reactors, construction elements, and the like.

The sphere of application of computers is expanding enormously. For this reason it is important to standardize computer hardware, improve human interaction with

computers, and develop domestic appliances with built-in microcomputers and minicomputers.

Without theory practice is blind. Soviet scientists today are ahead of foreign scientists in many aspects of theoretical cybernetics. The latest advances of theoretical science, along with the development of means to automate the processing of results from scientific investigations and experiments, should give rise to a new stage in creative scientific thought and accelerated introduction of the results of scientific research in practice.

Yu. V. Kapitonova, doctor of physicomathematical sciences.

Raising the efficiency of use of computers and computer software depends on the quality of personnel training. The production of computing machinery is developing very swiftly today, but the training of skilled computer users is still plainly inadequate. Work to improve the qualifications of computer users is one of the most important ways to raise labor productivity. At the same time we must raise the "intellectual" level of the computers themselves. It seems to me that computer-based artificial intellect systems that provide more comfortable conditions for human beings when they are given jobs now done by human beings can also be a means to raising the productivity of both physical and mental labor.

Information plays a very important part in the contemporary world. Artificial intellect systems as means of producing, processing, and transmitting information should become as necessary to people as the telephone, radio, and television.

Raising the "intellectual" level of computers inevitably involves making their functions, and therefore also their design, more complex. This requires that we set up industrial lines for automated production of computer components, that is, circuitry and program equipment. This presupposes, in turn, the development of the theory and systems for designing computers. The division of digital automata theory at the Institute of Cybernetics or the Ukrainian SSR Academy of Sciences is working on the development of such resources. We hope that the results of our research will be of benefit to the economy of the country before the 11th Five-Year Plan is over.

A. A. Letichevskiy, doctor of physicomathematical sciences.

Cybernetics science is developing in close interaction with applied problems, helping accomplish key national economic tasks related to the development and introduction of computers. Meeting the great challenges set down by the 26th Congress of the CPSU for Soviet society demands further improvement in computer hardware and creating new structures of computers and new methods of organizing computing processes to make maximum use of the potential of contemporary technology for manufacturing computer hardware.

The development and use of multiprocessor systems revived the theory of parallel computations, stimulating a transition from the study of abstract models far removed from any real systems to solving practical problems. The principle of macroconveyor organization of computations in multiprocessor systems, discovered

by V. M. Glushkov, led to a fundamentally new solution to the problem of efficient use of general-purpose systems with large numbers of processors. Further study and application of this principle is opening new prospects for the development of the theory of computing processes and drawing in related fields such as methods of numerical analysis, artificial intellect, and the like.

While working on applied problems, we must keep the prospects for development of fundamental knowledge in view. The algebra of algorithms, the theory of data structures, recursive computations, and other branches of the applied theory of algorithms constitute the theoretical basis for the developing field of mathematical science, which must work out and refine the general method of solving the problems of analyzing, synthesizing, and optimizing algorithms, identifying special classes of algorithms by putting abstract mathematical models in concrete form. As a specific example of such development we may point to the transition from studying flow charts of free data algebras to studying flow charts of algebras of data defined by systems of identities (abstract types of data).

The principal source of the problems of the applied theory of algorithms is experience constructing a system for automated design of the hardware and software of computers based on high-level languages that approximate the languages of mathematical definitiions and problem formulation. One of the promising lines of work in this field is the development of means and methods of theoretical set programming based on the formalized language of practical mathematics and including deductive means among the instruments used to design algorithms.

Yu. M. Yermol'yev, corresponding member of the Ukrainian SSR Academy of Sciences.

The changes that have taken place in stochastic programming in the last five years are remarkable. This relates above all to further development of direct stochastic methods of solving complex problems of investigating large-dimensionality operations. Efficient procedures have been developed to solve a broad range of problems of planning stocks, network optimization, agricultural production, two-stage problems of stochastic programming, and various other problems that have special structures of constraints. Solving practical problems necessitated a further expansion of the class of optimizable functions. Techniques of minimizing slightly convex, generalized differentiable, and Lipshits functions were investigated. A very general methodology of proofs was worked out to investigate the convergence of nonmonotonic optimization procedures.

Broad investigation of ε-quasigradient optimization techniques was done, and methods were constructed for solving minimax problems which did not require precise computation of the generalized gradient of the function of the maximum.

The applied problems of controlling industrial processes and dynamic and organizational systems where the structure of the object being optimized develops gradually led to working out numerical methods of nonstationary optimization.

Numerical techniques of solving stochastic problems of optimization by distribution functions that satisfy all common constraints were worked out.

There was intensive development of algorithms for solving stochastic minimax problems and problems with complex regression functions.

In connection with the development of packages of applied programs, great significance was given to investigating the effectiveness of numerical methods of nondifferentiable and stochastic optimization. The asymptotic behavior of stochastic optimization procedures in the class of convex large-dimensionality problems and resistance to random error in raw data were studied.

Numerical methods of nondifferentiable and stochastic optimization without computation of subgradients were developed. It was shown that in many practically important cases it is possible to substitute computation of stochastic finite differences for computation of generalized gradients. Finite difference procedures were developed which to a significant degree generalize known techniques of stochastic approximation and are applicable to solve complex stochastic minimax problems.

The Institute of Cybernetics of the Ukrainian SSR Academy of Sciences has now developed a package of applied programs for stochastic and nondifferentiable optimization which is designed to solve rough extremal problems of nonlinear and stochastic programming. The package functions under control of the YeS [Unified System] operating system and permits work in the automatic (batch), interactive, and multiprocess modes. Plans envision a second version of the package, oriented to solving conditional extremal problems, by 1985.

L. A. Kaluzhinin, doctor of physicomathematical sciences.

The development of computers and the broad opportunities for their application are especially important today, in the age of the scientific-technical revolution. The decrees of the 26th CPSU Congress envision concentrating efforts on solving such important problems as the development of mathematical theory, increasing its use for applied purposes, and improving computer technology, its basic elements and software, and means and systems for data collection, processing, and transmission.

The second half of the 20th century has seen the ever-expanding intrusion of mathematics into science and technology. Whereas in an earlier day the principal users of the results of mathematical theories were almost exclusively mechanics, physics, and the branches of engineering related to them, today mathematics finds application in chemistry, biology, economics, linguistics, and in the branches of cybernetic engineering — in the work of designing automatic and using automatic devices. This change and the expansion of its area of application have led to the formulation of a number of new problems in mathematics itself. The greatest change in focus within mathematics, we feel, is the increase in the importance of the fields of discrete mathematics. Whereas in the 19th Century and early 20th Century the analytic disciplines — differential and integral equations, variation calculus, and the others — were paramount for practical application, today algebra, combinatorial analysis, number theory, mathematical logic, and others play an equal role. The growth in the significance of these disciplines can be observed by the number of projects and publications.

A great deal of attention today is being given to solving problems of algebra and combinatorial analysis on computers. The problems that are arising are of great theoretical and practical interest.

I. N. Kovalenko.

With the development of cybernetics systems it becomes increasingly important to analyze and synthesize their efficiency and reliability with due regard for random factors. The theory of random processes, adapted to describing the action of real systems, and development of the analytic apparatus for investigating them are timely for solving these problems. Significant results have been obtained in these areas in recent years. Many concrete mass-service schemes have been studied, in particular those that take account of the specific characteristics of the operation of computers in computer systems. Attempts are underway to compose algorithms for calculating the characteristics of mass-servic systems within the framework of generalized mathematical schemes. The static modeling technique has acquired a large role and become the basic calculation method in systems research. Software systems for static system modeling have been devised.

The special significance of asymptotic techniques of calculating systems characteristics for systems research has been recognized in the last decade, chiefly thanks to the work of Soviet mathematicians. Thus, the theory of calculating highly reliable systems has been worked out and significant results have been obtained in the theory of consolidating the states of random processes. But all of these techniques require elaborate software for practical introduction. At the present time the corresponding packages of applied programs are being developed at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences.

Use of the capabilities of contemporary multiprocessor systems is a promising direction for systems research. Specifically, it is very convenient to realize static modeling algorithms on recursive-type computers because the organization of data exchange in such machines corresponds to the real processes being modeled.

Finally, we should note one more interesting area, the development of algorithms for deriving formulas for the characteristics of mass-service systems using computers based on the ANALITIK algorithmic language. The introduction of such techniques can save an enormous amount of "manual " work on formal conversions, which still take a great deal of time.

Our cybernetics scientists are studying many aspects of the theory of mass service and reliability and are making their contribution to the overall process of employing mathematics and cybernetics in our country's science and technology.

V. S. Korolyuk, academician of the Ukrainian SSR Academy of Sciences.

At the present time mathematical models that use the theory of random processes play a significant part in many diverse areas of cybernetic research. Therefore, the problems that arise in the theory of random processes and their applications will continue to be timely. Delimitation and development of the theory of those

classes of random processes that are most adequately applicable to analysis of complex stochastic systems is especially promising. The desire for an adequately complete mathematical description of the functioning of stochastic systems necessitates the use of increasingly complex mathematical models, and therefore also of more complex medical apparatus. In this way a two-sided situation arises: the more accurate the mathematical model is, the less possibility there will be of receiving practically efficient conclusions; at the same time, the simpler the mathematical model, the less reliable conclusions based on analyzing it will be. One of the effective ways to overcome this contradiction is to construct fairly precise mathematical models that take into account all significant aspects of the processes under consideration, with subsequent simplification of the model substantiated by mathematical theorems. Development of the method of phase consolidation of stochastic systems is an illustration of this approach. One of the timely problems of contemporary systems analysis is the development of similar methods of simplified analysis based on profound mathematical ideas.

O. S. Kulagina, candidate of physicomathematical sciences.

The problem of automating various types of processing of texts in natural language is one of the timely scientific problems involved in broadening the use of computer machinery. Among the diverse types of automatic text processing systems (for example, question-answer systems, systems to translate from certain languages to others, and the like) are language processors that analyze and synthesize texts. Language processors usually have three types of software: linguistic (vocabularies and grammars), mathematical-algorithm (algorithms and formalized representations of data and algorithms), and programs. The development of different types of software should proceed in a coordinated manner withir the framework of a single mathematical model that describes the structure of the natural languages and their functioning. The models developed at this point embrace chiefly morphology and syntax, but only partially the semantics of natural languages. Further progress requires fairly complete models of natural languages that take account of their specific features such as complexity because of the large number of objects with individual properties, indeterminacy of branches (transfers), the lack of precision with respect to many sets and relationships (including norms of correctness), the inequality of similar phenomena and constructions which is felt as a preference for certain ones over others and is difficult to express in authorize/prohibit terms, and so forth. Experience with the construction of systems for automatic text processing has shown quite clearly that the formal grammars proposed by N. Chomsky are not an adequate means of modeling natural languages and the external similarity between KS languages and natural languages only made a prompt understanding of this inadquacy more difficult. In the text analysis algorithms formulated on the basis of Chomsky grammars and constructed to recognize the affiliation of a phrase with a language generated by the formal grammar, syntactical structure occurs as a by-product of recognition. Such algorithms are unsatisfactory because of their instability with respect to deviations from grammatical norms. They do not give syntactical structure for phrases that do not correspond to their grammar, but in view of the above-mentioned properties of natural languages, the existence of phrases that do not correspond to the grammar is inevitable in a text, unless it is specially selected, with any grammar and they cannot be too infrequent.

In view of the above it would seem desirable for the journal KIBERNETIKA to publish articles on construction of mathematical models of the structure and functioning of natural languages, discussion of various approaches to this problem, discussion of more general or particular models, and description of various systems of automatic text processing and the functioning algorithms obtained using their results. Such work fits naturally in the journal's topic headings of mathematical models in linguistics and artificial intellect.

A. I. Kukhtenko.

It has long since become common knowledge that the ideas and methods of cybernetics are being applied not only to develop control systems for industrial facilities, but also that they are being used successfully in economics, biology and medicine, linguistics, ecology, and elsewhere.

The pure sciences (physics, chemistry, and theoretical biology) are a new arena which is being taken over somewhat slowly, but steadily by cybernetics. We are not speaking here of cybernetization of the experimental base, but rather of formulating and solving particular purely theoretical problems whose consideration requires the use of the ideas and methods of control theory (for example, investigating the stability of the process of controlling thermonuclear installations or elementary particle accelerators and development of quantum mechanics control systems).

The study of controlled objects and processes in these fields of knowledge leads naturally to the necessity of using mathematical means that are adequate to these objects and processes and this, in turn, requires uniformity (standardization) of the mathematical apparatus with which it is possible to describe both the object of control or process and the control system for it at the same time. All this requires the creation of new branches of control theory and leads to an expansion of the sphere of application of cybernetic methods and enriches the scientific basis of cybernetics itself.

As experience demonstrates, among the most promising areas of investigation that are fully adequate to the problems under consideration are the apparatus of Lee groups and algebras, differentiable manifolds, and externally differential forms as well as such branches of mathematical knowledge as the theory of layered spaces, the theory of special characteristics and accidents, the theory of jets, the theory of ambiguous images, and so on.

It must be recognized here that this is not simply a matter of substituting certain mathematical means for others; it is something more significant. As specialists in the field of scientific methodology say, it is a substitution of "paradigms," a replacement of the scientific basis. It occurred first in mathematics itself, and then effected a radical reorganization of such scientific disciplines as analytical mechanics and continuous medium mechanics as well as many branches of theoretical and mathematical physics, and now is gradually encompassing chemistry, theoretical biology, and also control theory. A great deal has already been done in this direction, but it also offers very broad prospects for further research on both the theoretical and applied levels.

B. N. Pshenichnyy, doctor of physicomathematical sciences.

The 26th CPSU Congress defined intensification as the principal line of development for our economy and industry. But intensification means maximal use of each process and each technology, and it is inconceivable without optimization of the use of all available resources. Therefore, development of optimization theory and fundamentally new computing techniques for optimization is becoming a pressing problem.

In the area of theory, I believe, it is very important to continue studying the problems which include nondifferentiable functions. There has been a trend in recent years to identify special classes of such functions, describe their properties, and construct the essential extremum conditions. This has made it possible to obtain essential conditions of a fundamentally new type which allow a very detailed characterization of extremum points in even the roughest problems. This is extremely significant for formulating optimization algorithms, even though it appears that narrower classes of functions must be identified when working out effective algorithms.

In connection with the problems of mathematical economics, the investigation of multiform representations, which can be used to describe economic processes, plays a large part. The steadily growing stream of publications in this field confirms this opinion. I also want to note that the problem of mathematical description of price formation in economic models is an important and extremely interesting matter. We are still far from an adequately complete solution to this problem.

The ultimate goal of the development of any theory is practice, and for optimization theory the practical outcome is effective new algorithms and packages of programs. Significant advances have been made in this in recent years. There has been further development of the techniques of sequential analysis of alternatives applicable to combinatorial problems and systemization, and new methods of nondifferentiable optimization have been worked out in addition to methods of solving general nonlinear programs of mathematical programming.

V. M. Glushkov's formulation of new principles and approaches to optimization problems relying on systems analysis was a major contribution to this problem area. These principles allow a different look at the methods by which problems are solved and make new demands of the developers of algorithms.

I will note in conclusion that, unfortunately, we still have not developed many tested program modules that can be used extensively by algorithm developers. Even for the basic problems such as linear programming and quadratic programming each organization uses its own programs, which other organizations do not have. Standardization of modules and universal distribution of them is one of the important practical challenges.

I. V. Sergiyenko, corresponding member of the Ukrainian SSR Academy of Sciences.

The materials of the 26th CPSU Congress devote a significant place to the problems of scientific and technical development, specifically the questions of the

development of automated data processing systems, automated control systems, and computer hardware.

The important role that mathematical methods of solving optimization problems of different types and computer and systems software are playing in the development of science and technology is common knowledge. Research in the field of development of the contemporary theory of optimization continues to be very timely. It is precisely rational use of mathematical methods of solving optimization problems that often makes possible significant improvements in the efficiency of automated control systems, computer-based automated data processing systems, automated design systems for complex objects, and the like when they are being set up.

In recent times there has been intensive development of discrete software and its varied applications. This is because many problems of control, planning, and design are described by discrete programming models. In the near future effective techniques are to be worked out for solving large-dimensionality problems of this type and the software of packages of programs that will permit solving problems of this class in the interactive mode will be developed. It is interesting here to work out both precise and approximate methods of solving these problems, and it is equally important to develop software for realizing them on computers which will allow dynamic changes in the mathematical models of the problems being solved and the parameters of the algorithms used during this in different modes when necessary.

Several other questions continue to be timely: evaluating the quality of optimization methods and mathematical means of realizing them in the development stage; working out new approaches to formal description of important national economic problems; and, building automated means that allow efficient investigations of complex systems.

S. L. Sobolev, academician.

In recent years I have been working on the theory of cubic and quadrature formulas. This work relates to a field bordering on classical mathematical analysis. I am interested in questions of optimizing such formulas for various classes of differentiable functions and in the asymptotics of optimal formulas and norms of the functional of error in different situations. To the present time these problems have been formulated in classical Banach and even Gilbert space.

It seems to me that it would be extremely interesting if one were to succeed in giving an analytic interpretation of variation problems of the minimum norm of error with complementary constraints on permissible formulas. For example, to find a convenient solution to the problem of optimal formulas with moduli of coefficients constrained in advance.

A simpler problem, but also one that needs to be worked out, is the asymptotics of optimal (but not just asymptotically) cubic formulas.

E. F. Skorokhod'ko, doctor of philological sciences.

The second half of the 1970's in our country saw widespread use of document information retrieval systems based on program packages such as ASPID, ASOD, and POISK-1. Despite the obvious advantages of these systems — standardization of software, a high level of technological suitability, well-developed service facilities, and so on — they are not without a number of shortcomings (many of which, strange as it may seem, were not found in the experimental systems of the 1960's). Thus, the generally laudable desire for maximum simplification of linguistic facilities led to a situation where the volume of nonautomated operations in processing input data was unjustifiably large. Indexing queries is a particular burden. It is this operation in the systems under consideration that permits partial compensation for the weakness of linguistic facilities, in particular the lack of a well-developed system of paradigm relationships and, as a consequence, a diminution of the completeness of the search and a low level of adaptability to user interests. In addition to the fact that it requires highly qualified service personnel and takes a great deal of time, in many cases the necessity of manual indexing of queries makes it impossible to organize direct dialogue with the user.

Indexing documents and compiling information retrieval thesauruses are also very labor-intensive operations, despite the fact that contemporary packages of applied programs for document systems to some extent ease this job by permitting automatic indexing of short texts of the abstract and annotation type and offering the possibility of obtaining a vocabulary of the indexed texts.

In the current five-year plan the efforts of scientists and developers of document information retrieval systems should be directed primarily to solving these problems, that is, to setting up efficient systems for automatic text reference and indexing queries and documents. In addition they should work to set up automated systems to design thesauruses for information retrieval systems and data banks which can select words, on the one hand, and establish paradigm relationships among descriptors and structure the thesaurus, on the other.

Ye. L. Yushchenko, corresponding member of the Ukrainian SSR Academy of Sciences.

The 26th CPSU Congress outlined the impressive prospects for development of our domestic science and technology. Cybernetics, in particular, was given the challenges of further raising the productivity of computers, improving their material support, and expanding the sphere of practical application in industry and the national economy. It is essential to intensify research on development of programming languages, translating, interpreting, and operating systems, packages of applied programs, data bases, and other components of the software of contemporary and future computers. Within the framework of the further development of means of human interaction with the computer, the problem orientation of languages using contemporary programming technology, the apparatus of macrogeneration, standardized base languages, and control systems for data bases is becoming particularly timely. Interesting results have been received in this area for the language COBOL. It is also very timely to develop parametric programming systems, to put together sets of packages of applied programs, and to

set up systems to teach languages and programming techniques. In connection with the appearance of multiprocessor computing systems it becomes interesting to develop parallel programming languages and systems, means of organizing the computing process in multimachine computing systems, and so on.

Shaping the mathematical foundations of programming technology, including parallel programming, is an important line of study in theoretical and systems programming.

Interesting results have been obtained in the schematology of structural programming, the theory of parametric models of languages, and formal methods of structural program design. Work is underway to devise structural programming hardware and use it to solve various practical problems. Within the framework of the further development of this area there may be progress in research on the analysis and synthesis of structured programs, their verification, transformation, and documentation, organizing the labor of the collective of developers, and the like.

It should be emphasized that solving the problems of program compatibility among computers, storing libraries of programs, and preparing users at different levels of qualifications by means of well-developed teaching systems will promote a qualitative advance in the development of the "intellect" of coming generations of computers.

A. P. Yershov, corresponding member of the Ukrainian SSR Academy of Sciences.

The lasting significance of the 26th CPSU Congress lies in the fact that it marked a radical turning point in our national economy, from the extensive way of development to the intensive task. In a historically short period of time our country has resolved the great problem of developing its natural resources, creating up-to-date productive forces, shaping the essential educational and cultural potential of the population, and insuring full participation by the population in public production. Further growth in production is no longer possible by simply bringing in new natural and human resources. It requires rapid incorporation of scientific-technical advances, fuller use of creative potential, and efficient management.

One of the conditions of meeting this new historic challenge is bolstering the scientific principle in the organization of public production and management. This finds specific expression in the need for more thorough knowledge and comprehensive consideration of the objective fundamental laws of developed socialism. The complexity of production relationships and the limited character of resources intensify the interdependence of the different parts of the economic mechanism and the impact of particular decisions on the final results. Intuitive, approximate, or local knowledge of these laws will no longer provide the necessary efficiency. It must be replaced by deep, broad, and advanced knowledge and the ability to apply it in the interests of achieving the final objective at the proper time.

It is deeply impressive, how much the goals of the 26th Congress reflect the challenges facing our computer science. Right from the start and until just recently programming has developed extensively, embracing new areas of computer

application, formulating new methods, and multiplying essential features. We have all kinds of programming: theoretical, applied, systems, recursive, structural, parallel, functional, systematic, and applicative. We distinguish operational, mathematical, denotational, transformational, logical, and reduction semantics. By itself the number of such terms, and there is a whole field of study behind each one, illustrates the diversity of the directions of growth in programming.

In recent years, however, it has become clear that simply adding accumulated knowledge will never be sufficient to overcome the steadily growing barrier of complexity raised by the practical problems of programming. Indeed, in science itself the division into new fields is increasingly moving to the classical spiral of development where essential similarities reveal themselves in the most unexpected places and things that were new are already obsolete.

The accumulation of phenomena such as this forces us to think that the problem of bolstering the scientific principle in programming, making the transition from extensive development of computer science to intensive, and learning and using the deep relationships between programming and other sciences and types of human activity is a problem that not only should, but also can be made the focus of further progress in computer affairs. With a view to supporting this statement I will give three examples of the recently recognized relationship between certain fundamental concepts of mathematics and programming.

Example 1. Church-Rosser systems and program processors [1]. In the 1930's the so-called lambda calculus of Church played a large part in the formation of the theory of computability. The fact that this calculus was equivalent in capacity with the concept of the recursive determination according to Erbran and Goedel enabled Church to formulate his famous thesis and give the first example of a concrete, algorithmically insoluble mass problem. Lambda computations proved very inconvenient in use, so further development of the theory of computability used other models of computations. At the same time lambda computations have a number of remarkable characteristics. One of them, which has come to be called the Church-Rosser property, consists in the following. The lambda calculus may be viewed as a system of text transformations. Individual transformations are called reductions (in classical calculus there is just one transformation, of the substitution type: $\lambda x f(x)$, $x = c \to f(c)$). Among the different forms of text the so-called normal form, which is a fixed point for all reductions, stands out syntactically.

The Church-Rosser property consists in the fact that if we transform initial text t into normal form t_n, it will be one and any intermediate form of text t is also reduced to t_n. The systems of transformations that satisfy this property came to be called Church-Rosser transforms. These transforms are the ideal means for describing the nondeterministic processes of manipulations with different constructional objects. It is regrettable that with a straight-line approach conventional processors were very far from possessing the Church-Rosser property.

Programming traditionally considered representations of the type $P \times D \to D$ (computations of programs from P on data from D) or $P \to P$ program transformations.

Study of the former was the subject of semantics, while the latter was studied by the theory of flowcharts. It took a great deal of time (some of it to ovecome the preferential bias for considering computer models with invariable programs) to discover the usefulness of studying representations of the type $P \times D \to P \times D$ (program processing, mixed computations, and the like). One of the most interesting results of this approach is that the semantics of simple programming languages described in a system of base transformations of the type $P \times D \to P \times D$ can be adequately represented in the form of a Church-Rosser transform.

This was established first for recursive programs, and then a little later for structured operator programs with memory as well. This observation rehabilitates the operational approach to describing semantics, liberating it from the over-definitiveness of the purely deterministic approach, and makes it possible in the context of representations $P \times D \to P \times D$ to develop a general theory of very different program processors, which formerly had been considered separately.

Example 2. Mixed computations and the s-m-n-theorem [2]. Suppose program $p(X, Y)$ computes function $f(X, Y)$. It is possible to formulate the problem of the search for a systematic procedure of type $P \times D \to P \times D$ which, from program $p(X, Y)$ and the assigned value of one of the independent variables (suppose $X = a$), would construct its projection, that is, program $p_a(Y)$, which computes function $\phi(Y) = f(a, Y)$. Such a procedure $M(P, X, Y)$: $M(p, a, Y) = p_a(Y)$ in programming language $L = (P, D)$, $D \supseteq X \times Y$, distinct from the conventional computer $V(P, X, Y)$: $M(p, a, b) = p(a, b)$, is naturally called a mixed computer for language L.

In the course of the 1970's several programmers independently made an important observation which greatly influenced ideas of the nature of translation in comparison with the previous 20 years of development of this important branch of programming. Suppose that in processing language L mixed computer mix (P, X, Y) for this language and interpreter int(π, ξ) of a certain input language L are programmed. In the mixed computation mode we will apply int to the input program π, mix to the interpreter int and then mix to itself. It turns out then that the projection of the interpreter on the input program gives its object code in language L, while the projection of the mixed computer on the interpreter of the input language gives a translator from this language and the projection of the mixed computer on itself gives a translator of translators in the processing language. It is difficult to ignore such a fundamental and meaningful determination of the basic objects of translation theory.

Another interesting feature of this story is the fact that logicians knew the concept of the mixed computation and its role in translation at a time when programming was just getting on its feet. Already in the 1940's S. K. Klini knew the so-called s-m-n theorem (given below in a formulation for $m = n = 1$): there is a numeration G of partially recursive functions with universal functions $U^1(p, x)$ and $U^2(p, x, y)$ for one-place and two-place functions respectively as well as the primitively recursive function $s(p, x)$ such that for any two-place function $\phi(x, y)$ with number G_ϕ. The following is correct.

$$\phi(x, y) = U^2(G_\phi, x, y) = U^1(s(G_\phi, x), y).$$

It is apparent that the s-m-n-function corresponds exactly with the mixed computer for the programming language in terms of partially recursive functions. In the 1950's V. A. Uspenskiy worked on the theory of numerations, which was presented in his well-known "Lectures on Computable Functions." A numeration is called a main numeration if any other computable numeration is translated in a generally recursive manner within it. V. A. Uspenskiy proved the theorem in which he actually related the property of a computable numeration being a main numeration with the feasibility of the s-m-n-theorem for it.

It is interesting to observe that some logicians consider the s-m-n theorem one of the most powerful tools of the theory of computability, comparable in productivity to the Church thesis.

Example 3. Computable functions and automatic determinants [3]. The period of development of the foundations of mathematics from the early 1930's until the end of the 1960's was characterized by a large number of different definitions of the computable function and the algorithm. We will mention only the best-known ones: representability according to Goedel, recursive definability of Erbran and Goedel, lambda definability according to Church, computability according to Turing, partial recursiveness according to Klini, canonical deducibility according to Post, normal algorithms according to Markov, discrete transforms according to Glushkov,[1] operator algorithms according to Yershov, recursive programs according to MacCarthy, and Diophantine computability according to Matiyasevich. Behind this apparent diversity of conceptions and premises there is in fact one of the most immutable and absolute, as specialists put it, concepts of mathematics — the concept of the effectively computable functions. The absoluteness of this concept finds expression in the fact that each of the above definitions describes precisely one and the same class of arithmetic functions. On the other side from this mutual reducibility of the different definitions of computability is the unsettling circumstance that each of these definitions is developing, so to speak, its own theory. A basic question arises: is it possible to give an abstract definition of computability which would have the same status in relation to all the concrete theories, making them in some sense models of this abstract definition.

The first cycle of research in this direction showed that the initial step of abstraction should be a search for a definition of computability for random subject areas D with fixed sets of elementary operations $\Phi = \{\phi_1, \dots, \phi_n\}$ and predicates $\Pi = \{\pi_1, \dots, \pi_m\}$, that is, for abstract algebraic systems $\mathfrak{U} = D, \Phi, \Pi$.

[1]The author apparently does not consider certain aspects of the theory of discrete transforms from which it should better be classified with the abstract theories of computability considered in the third example than with the concrete, refinements of the concept of the computable function (from the editors).

Further searching for an abstract definition of computability, however, bogged down in the stage of formally relating the abstract theory to computability in a natural series or in the stage of constructing axiomatic theories that postulate the strongest properties of the class of computable functions, in conventional theories obtained, so to speak, at the very end. The weakness of the first approach was that the abstract theory is not in the same position in relation to the concrete theory, but rather is constructed on the basis of one of them. With the second approach, the description of a concrete theory as a model for the abstract one proved conceptually comparable to direct development of a concrete theory.

A new approach to defining abstract computability took shape on the basis of knowledge accumulated in work on theoretical programming. One of the fundamental concepts of programming and the theory of computability is the concept of the protocol, that is, the set of operations performed and decisions made in receiving a result corresponding to an assigned independent variable. The protocol is naturally represented in the form of a certain text in the alphabet of the symbols of input variables, elementary operations Φ, and predicates Π together with their truth values. The set of protocols corresponding to real computations is not very interesting. It is an enumerable set that is adequate to the graph of the function. It is a different matter if we consider flowcharts or abstract programs, that is, if we do not resort to interpreting the abstract algebraic system $\mathfrak{U}$. Instead of the process of computations, in these charts it is possible to give a generating process of a very simple nature which constructs a certain set of formal protocols, that is, a language in the above-mentioned alphabet. This set has the following properties: (a) each protocol for a given interpretation is either contradictory or gives a value that belongs to the graph of the function; (b) two programs with identical sets of formal protocols complete the same function with any interpretation.

These properties enable us to call the set of formal protocols the determinants of computable functions. The most important property of the determinants for flowcharts is that these determinants are very simple automatic sets: the determinants for Yanov charts are interpreted as finite automata, while for random statement charts they are two-band finite automata and for recursive charts they are context-free languages (a nonfinite result).

These properties of the determinants lead us to take the determinants as the basis of the definition instead of detecting them in concrete computing models.

The definition. A function is computable in the algebraic system $\mathfrak{U} = D, \Phi, \Pi$, if there exists for it a certain automat set of formal protocols (determinants) such that any significant protocol from the determinants is a certain value of the function and, conversely, for any value of the functions there is at least one significant protocol that gives this value.

Work to develop the theory of computability on the basis of this type of definition is just beginning, but already its abstract character can be seen: abstractions from the syntax of the program and from the concrete general-purpose algorithms; reliance on the concept of the automat set, an object of

much simpler nature than primitively recursive functions; the basic, at this time, language of system programming in the theory of recursive functions; the strict division into the combinatorial and properly computing aspects of the work of the algorithm.

Let us summarize what we wanted to say with these three examples which are, it seems to us, very typical for evaluating current trends in programming. The development of a science, even though it is aimed at attaining objective knowledge, is in itself a social process and therefore reflects a number of general rules inherent in such a process. One of these rules, specifically the expression of the formation's maturity in the turning point from extensive to intensive development, is convincingly confirmed in these examples. A second, equally instructive observation is that an idea, even when expressed, does not by itself win over the masses. In addition to everything else, it must be expressed in time to be heard and there must be means to implement it. P. Landin noticed the connection between the Algol-60 and lambda calculus almost immediately after it appeared, but more than 10 years passed before the development of applicative languages and the transformation approach to programming filled in the situation to the point where the productivity of these ties could be seen, above all the approaches to further elaboration of Church-Rosser systems. Similarly, translation theory would not have come to the concept of the mixed computation if the almost insuperable difficulties of effective realization of "big" programming languages had not forced an almost desperate desire for a more general and all-encompassing view of translation.

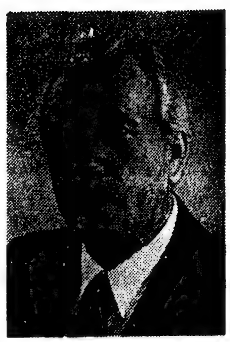
On the other hand, we can also detect the lasting importance of the subjective human factor in the development of a scientific discipline. The distances between sciences and discoveries in various areas are only overcome by people capable of understanding the languages of each side. The synthesis and interpenetration of the sciences, mediated by the active, personal position of the scientist, is the only way to resolve one of the most dramatic contradictions of the process of acquiring knowledge: the unity of the world and the lack of coordination and restricted competence of the specialists. It is, incidentally, one of the principal missions of our journal to overcome this contradiction.

V. M. Glushkov, academician and editor-in-chief of the journal

Yu. V. Kapitonova, doctor of physico-mathematical sciences

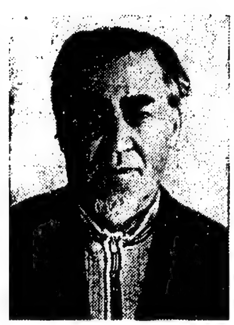
Yu. M. Yermol'yev, corresponding member of the Ukráinian SSR Academy of Sciences

V. S. Mikhalevich, academician of the Ukrainian SSR Academy of Sciences and deputy chief editor



A. A. Dorodnitsyn, academician

A. A. Letichevskiy, doctor of physicomathematical sciences

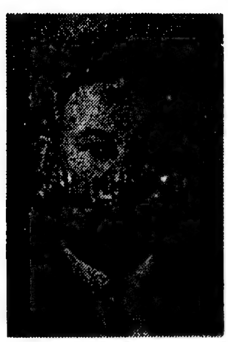


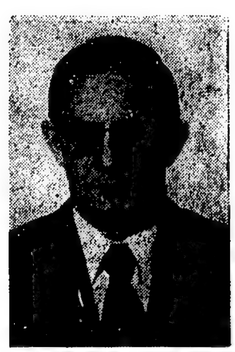
L. A. Kaluzhnin, doctor of physico-mathematical sciences

O. S. Kulagina, candidate of physicomathematical sciences

I. N. Kovalenko, corresponding member of the Ukrainian SSR Academy of Sciences

A. I. Kukhtenko, academician of the Ukrainian SSR Academy of Sciences

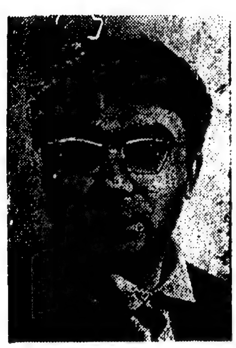




V. S. Korolyuk, academician of the Ukrainian SSR Academy of Sciences

T. P. Mar'yanovich, candidate of physicomathematical sciences

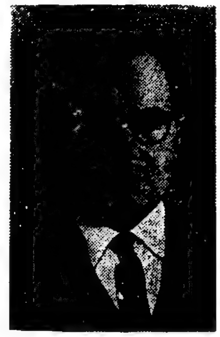
I. V. Sergiyenko, corresponding member of the Ukrainian SSR Academy of Sciences



B. N. Pshenichnyy, doctor of physicomathematical sciences

S. L. Sobolev, academician

E. F. Skorokhod'ko, doctor of philological sciences



A. P. Yershov, corresponding member of the Ukrainian SSR Academy of Sciences

Ye. L. Yushchenko, corresponding member of the Ukrainian SSR Academy of Sciences

Yu. I. Zhuravlev, doctor of physico-mathematical sciences

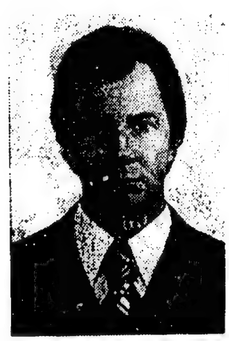
D. A. Suprunenko, doctor of physicomathematical sciences

N. N. Moiseyev, corresponding member of the Ukrainian SSR Academy of Sciences

N. Z. Shor, doctor of physicomathematical sciences

V. A. Statulyavichyus, academician

V. L. Volkovich, doctor of technical sciences



V. A. Kovalevskiy, doctor of technical sciences

I. V. Romanovskiy, doctor
of physicomathematical sciences

N. S. Furs, deputy
chief editor

11,176
CSO: 1863/16

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000500010023-0

UDC 681.3.004(075.3)

COMPUTER SERVICING AND REPAIR FUNDAMENTALS

Moscow OSNOVY EKSPLUATATSII I REMONTA EVM in Russian 1981 (signed to press 9 Feb 81) pp 2-5, 229, 232-235, 248-252, 278-279

[From book "Computer Servicing and Repair Fundamentals", by Igor' Alekseyevich Orlov, Valeriy Fedorovich Kornyushko, Valeriy Viktorovich Burlyayev and Vladislav Nikolayevich Avdeyev, Energoizdat, 40,000 copies, 280 pages]

[Excerpts] Questions relating to testing and diagnosis are discussed, as well as the general principles of the organization of preventive and repair work in the process of the equipment servicing of computers. The general structure of a system for testing the operation of computers is presented along with the most widely used codes for detecting errors and a description is given of the organization of hardware, test and software-hardware inspection of computers. A technical and economic evaluation is made of measures for the servicing of computers.

Intended for students at secondary special educational institutions. Can be helpful to a broad range of engineers and technicians interested in questions relating to the maintenance and repair of computers.

Foreword

This textbook is oriented toward the course "Fundamentals of Computer Servicing and Repair" for students at secondary special educational institutions majoring in "Computers, Electronic Instruments and Equipment."

The extensive introduction of computer technology in various sectors of the national economy has necessitated the organization of the correct and efficient servicing of computers.

By the servicing of such a complex dynamic system as a modern computer is must be understood the process of using a computer for its purpose while maintaining the computer in good working order. This continuous process consists of various measures: scheduled maintenance, restoration of serviceability after failure, periodic preventive work, etc. The accomplishment of these measures requires considerable expenditures, and these expenditures depend to a considerable extent on how efficiently these measures have been organized.

As the range of application of digital computers expands, the job of guaranteeing reliability of their performance takes on ever more importance. Meanwhile the

complication of computers, the increase in the number of operating modes for them and the use of a new element base--integrated circuits--have made it difficult to test the correctness of their operation and to find malfunctions in a computer.

A great number of methods exist for the technical diagnosis of computers, utilizing both added testing equipment and purely software for the automatic diagnosis of computer malfunctions. The history of the development of methods of diagnosing computers has shown that first- and second-generation machines were serviced by very experimental technicians who for the purpose of finding malfunctions used basically intuitive search procedures based on their experience in working with a specific machine. It is rather difficult to gain this experience and pass it on when working with modern computers. In addition, the difficulty of hardware diagnosis is intensified by the fact that it is impossible to observe signals directly in internal points of the circuit in integrated circuits and the addition of numerous check points is clearly inefficient.

In connection with this software and hardware-software methods of testing the serviceability of a computer are under intense development at the present time and are taking on ever greater importance.

Taking into account the complexity of the question presented and the general direction of the course, the authors did not set themselves the goal of writing a manual with a detailed description of all methods and approaches for automatic testing, servicing and repair of computers. The goal of this book is to discuss the basic methods of testing and servicing computers in general, and in sufficiently complete a form as possible using as an example modern third-generation computers included among YeS [Unified Series] computers. In discussing specific examples of procedures for testing individual units and pieces of computer equipment, tests for a program check and recommendations on the servicing and repair of computers, the authors have endeavored to discuss solutions which are general for Unified Series computers (e.g., the TYeST-monitor [TEST-monitor] test monitoring system is a general system for the majority of these computers). In cases when this is impossible, examples have been presented for one of the most advanced and widely used medium-capacity YeS computers--the YeS-1022. Since this textbook is intended for students who in the future are to take part directly in solving problems and servicing computers under the guidance of servicing documentation, specific test examples, circuitry solutions and servicing recommendations have been discussed on the basis of instruction materials for the equipment servicing of YeS computers. However, this in no way reduces the generality of the conclusions and recommendations presented in this book, since the methods for the technical diagnosis and repair of various third-generation general-purpose computers are quite similar.

Work on this book was divided between the authors in the following manner: Chs 1-5 were written by I.A. Orlov, sec 5.6 by V.N. Avdeyev, chs 6 and 7 by V.F. Kornyushko and chs 8-11 by V.V. Burlyayev.

The authors wish to express their sincere gratitude to the reviewers--the team at the computer and software department of Moscow Institute of Railway Transportation Engineers--as well as to Candidate of Technical Sciences V.G. Persheyev, an assistant professor in this department, and to Moscow Radio Engineering Technical School imeni A.A. Raspletin Teacher K.A. Neshumova and to the science editor of this book,

Candidate of Technical Sciences B.N. Sevryukov, who helped to improve the manuscript with their comments and helpful advice.

A promising form of equipment servicing is the centralized servicing (TsTO) network, which makes it possible to solve the problem of spare parts and the shortage of technical personnel. The organization of complete centralized servicing (warranty repairs, periodic work and the like) makes it possible to solve many problems relating to the servicing of computers, to accumulate statistics on the failures of various pieces of computing equipment and units, and to improve the reliability of computers as a whole.

The performance of TsTO work is carried out on the basis of business contracts. The approximate cost of a TsTO preventive inspection for various computers is given in table 9-1. In the future the cost of TsTO services will be lowered and their list will be expanded.

Table 9-1. Cost of TsTO Preventive Inspection

Type of computer	Wholesale price of preventive inspection for a single computer depending on the number of them in the organization, rubles					Cost of overhauling a single computer, rubles
	1	2	3	4	5	
"Minsk-32"	5510	4460	3840	3630	3510	7730
M-220, M-222	7800	6290	5790	5550	5400	7730
"Ural-11," "Ural-14," "Ural-16"	12100	10100	9420	9090	8890	11500
BESM-6	18200	15100	14100	-	-	22000
YeS-1020	13300	10700	9850	9400	9200	11500
YeS-1030	18500	15000	13800	13200	12800	22000
"Nairi-2, S"	1360	880	710	600	580	1640

An example of the weekly schedule for four magnetic disks, one of which is a spare, is given in table 9-3.

Table 9-3. Weekly Schedule of Preventive Maintenance for Magnetic Disk Storage Units

Unit	Plant number	Day of week						
		1	2	3	4	5	6	7
5056		PR						
5056				PR				
5056						PR		
5056			R		R		R	PR

Note: PR--preventive maintenance; R--spare

Network planning and control methods are often used for drawing up a schedule of preventive work with the presence at a computing center of a great number of computers and other hardware.

9-3. Quantitative Evaluations of Preventive Measures

One key characteristic of preventive maintenance is the duration of computer preventive maintenance:

$$t_{\text{проф}} = \sum_{i=1}^{k} t_{\text{п}i} + \sum_{j=1}^{n} t_{\text{в}j} + t_{\text{ф.к}},$$

where

$$\sum_{i=1}^{k} t_{\text{п}i}$$

is the total time for the performance of k preventive measures performed in succession;

$$\sum_{j=1}^{n} t_{\text{в}j}$$

is the time for the restoration of n malfunctions during the preventive maintenance period; and $t_{f.k}$ is the functional testing time.

It must be mentioned that the level of skills of service personnel influences to a great extent the duration of preventive maintenance. It is possible to reduce considerably the total time of failures and the total expenditure of time for carrying out preventive work by raising the level of technical training and by generalizing and introducing advanced know-how relating to servicing.

Values of the key performance indicators of a YeS-1030 computer during three years of use are presented in table 9-4. As is obvious from this table, in the third year of use, on account of the accumulation of know-how and improvement of the quality of servicing, it was possible to reduce considerably the total time for preventive maintenance while increasing all remaining indicators of the computer's performance. It can be seen in table 9-5 how the total time for failures of individual units of this computer changed year by year. Some increase in the time for failures of peripheral units of the YeS-1030 in the third year of use testifies to the rapid wear of these units and their low reliability.

An analysis of statistical data on the use of a specific computer makes it possible to give recommendations relating to substituting frequent preventive maintenance with less frequent preventive maintenance (e.g., daily with weekly). This makes it possible to increase the time for using a computer directly for computing work.

The degree of influence of preventive maintenance on the reliability characteristics of a computer can be expressed in terms of the increase in mean time between failures:

$$\Delta T_o = T_{o.pr} - T_o ,$$

where $T_{o.pr}$ is the mean time between failures for the computer with preventive maintenance and T_o is the mean time between failures for a computer which has not been given preventive maintenance.

Table 9-4. Key Performance Indicators of YeS-1030 Computer

Performance indicators of computer	Value of indicators by year of use, hours		
	First	Second	Third
"On" time	3974	6865	7251
Effective time	3220	5560	6223
Preventive maintenance	363	602	582
Failures and malfunctions	390	500	353
Mean time between failures	21.3	45.7	97.3
Mean restoration time	2.8	1.6	1.0
Mean-24-h effective time	8.5	15.2	17.1
Mean-24-h utilization	11.0	18.9	19.8
Mean time of uninterrupted operation	4.8	7.8	9.7
Efficiency without taking into account preventive maintenance (percent)	92.2	92.8	94.6

Table 9-5. Failures of YeS-1030 Computer Units

Unit	Type of unit	Failure time by year of use, percent		
		First	Second	Third
Processor	YeS-2030	11.7	8.1	6.2
Channels	YeS-4430	11.4	10.8	9.8
Working storage	YeS-3203	7.5	2.0	2.0
Disk storage	YeS-5052	17.0	20.2	33.8
Tape storage	YeS-5010	9.5	7.2	12.2
Control units	YeS-5551 YeS-5511	3.1	3.0	6.1
Alphanumeric printer	YeS-7030	9.6	8.3	5.2
Input unit	YeS-6012	10.4	11.4	7.8
Operator's console	YeS-7077	13.3	17.2	6.1
Ouput punch	YeS-7010	5.5	11.3	10.5
Power supply racks	-	1.0	0.5	-

The relative value of the increase in mean time between failures equals

$$\delta T_o = \Delta T_o / T_o = T_{o.pr} / T_o - 1 .$$

The ratio $w = T_{o.pr}/T_o$ represents the effectiveness of preventive maintenance.

Another important quantitative characteristic is the preventive maintenance effectiveness factor, k_{prof}, which characterizes the degree of failure-free performance of a computer on account of the prevention of failures at the moment of preventive maintenance.

Let us introduce the following symbols: n_{prof} is the number of failures discovered during preventive maintenance; n_o is the number of failures which originate in the process of operation of the computer; $n_{obshch} = n_o + n_{prof}$ is the total number of failures of the computer during the period of use.

Then

$$k_{prof} = n_{prof}/n_{obshch} ,$$

and on the assumption that the flows of failures in the computer are the most simple,

$$w = n_{obshch}/n_o ,$$

or

$$w = 1/(1-k_{prof}) .$$

Thus, these expressions make it possible to give a quantitative evaluation of the degree of improvement of the reliability of a computer on account of preventive maintenance.

Now let us discuss the organization of preventive maintenance of units for controlling magnetic tape storages, YeS-5517's, and magnetic disk storages, YeS-5551M's.

The kinds of preventive maintenance for YeS-5517's are: 1) 24-hour (duration of 25 min), 2) biweekly (2 h), 3) monthly (4 h) and 4) semiannually (8 h).

Only semiannual preventive maintenance for a period of 2 h is called for for a YeS-5551M unit.

In 24-h maintenance of a YeS-5517 unit an external inspection is made and external parts are dusted, and a check is made of power supply unit voltages.

In weekly preventive maintenance a check is made of the operation of the storage, and in monthly of all storages according to the YeS-5517 test program while changing the voltage of ± 5-V power supplies in steps over the range of 4.75 to 5.25 V.

In addition, a check is made of the power supply overload protection system.

Semiannual preventive maintenance includes all monthly preventive maintenance steps, as well as a check of the time parameters of write and read synchronizing generators, the delay counter and stable signal conditioners. Time intervals are measured at check points specified by the servicing instructions, by means of a type Ch3-12 digital frequency meter. These measurements are made while altering the power supply voltage by five percent.

Practically all measures provided in the semiannual preventive maintenance of a YeS-5517 are carried out in performing semiannual testing and preventive maintenance work for the YeS-5551M unit.

The maintenance of YeS computers has made it possible to reveal the most typical reasons for failures of key units and to give recommendations on eliminating them:

Processor and channels--breaking of wiring connections, failure of contacts and demountable standard replacement elements, poor-quality wiring in racks. Elimination--complete cleaning of wiring, regular washing of TEZ [standard replacement element] connector assemblies with alcohol.

Main storage--wiring errors, poor debugging of memory matrix units (BZM's), failures of TEZ's in heavy-use circuits. Elimination--cleaning of wiring, debugging of BZM's, bench testing of TEZ's.

Magnetic disks--short service life of magnetic heads, power supply failures, positioner failures, infringement of write and read compatibility. Elimination--thorough maintenance of airtightness, cleanliness and humidity with complete shielding from the influence of electromagnetic radiation; careful adjustment and fitting of parts.

Punched card input unit--poor reliability of illumination lamps in readout unit, high sensitivity to quality of punched cards. Elimination--careful control of quality of punched cards, obligatory output of information onto the display's screen for monitoring purposes.

YeS-7030, YeS-7010 and YeS-7022 data output units--poor quality of inked ribbon, low reliability of punch. Elimination--use only nylon-base tape, replace punch with another type.

Chapter 11.

Technical-Economic Evaluation of Measures for Computer Servicing

11-1. Selection and Justification of Criteria for the Effectiveness of the Utilization of Computers

Keeping computers in good working condition and their effective utilization at minimum cost require intelligent organization of maintenance of the entire computing system.

The need for a detailed economic analysis of the utilization of computers is dictated by rapid advances in computer technology accompanied by enormous financial input and rapid changes in the structure of this input (the percentage of input for electronic components, software, storage and peripheral units, terminals, communications lines, etc.).

The most important technical and economic characteristic of a computing system is the productivity of the computer. The productivity of a computer reflects not only the technical characteristics of the central part of the computer (the speed of response of the processor, storage units and input/output channels), but also the

characteristics of external units and operating systems and even the efficiency of the specific configuration of the computing system in solving various problems.

The productivity of computing systems depends to a great extent on the technical principles of their implementation, i.e., the principles of the structural organization and functioning of hardware and the operating system of the computer.

Of great importance for improving the economic efficiency of a computer is improvement of the labor productivity of the programmer. The labor productivity of a programmer depends not only on his or her skills and the complexity of the problem, but also on the mode in which the computer is used.

The mode of use in this case determines the principle of the interaction of the programmer with the computer in the process of debugging the program.

The quantitative expression of the indicator of the productivity of a computer causes certain difficulties. This is due to a great number of factors influencing the total productivity of the computing system and to their complex interaction. In evaluating the productivity of a computer usually a quantitative analysis is made of the influence of various factors on the total productivity of the system.

In connection with the difficulty of estimating an absolute indicator of productivity it is more convenient to use an indicator of the relative productivity of a given i-th computer in relation to a base j-th model:

$$I_{ij} = P_i/P_j ,$$

where P is the productivity of the computer.

A distinction is made between the nominal, complex and combined productivity of a computer.

By nominal productivity, P_n, is meant the productivity of the processor, measured by the amount of operations per unit of time for a typical mix of procedures (cf. ch 2).

The complex productivity, P_k, is the productivity of the computer taking into account the work of the input/output units, storage units and other additional equipment.

Indicators of combined productivity are determined as follows:

$$P_{s.n}\ [\text{combined nominal}] = P_n k_{t.i}\ ;\quad P_{s.k}\ [\text{combined complex}] = P_k k_{t.i}\ ,$$

where $k_{t.i}$ is the coefficient of equipment utilization for the computer, determined by the equation in ch 2.

In calculating nominal productivity part of the parameters which exert a considerable influence on the computer's operation are left out of consideration (capacity of working storage and external storage units, operating speed of peripherals, etc.).

Nominal productivity is calculated under ideal conditions of the computer's operation. In actuality it is impossible to achieve this productivity since in the single-program mode time is lost on idle time in the input/output of information and in the multiprogram mode and time-sharing mode system time losses exist.

The change in indexes of the nominal and complex productivity of a YeS-1020 and "Minsk-32" computer in relation to a "Minsk-22" computer as a function of ρ -- the percentage of economic problems in the computer's total load--is illustrated in fig 11-1. It is obvious from this figure that the efficiency of a YeS-1020 computer is considerably better than that of the "Minsk-32" second-generation computer.

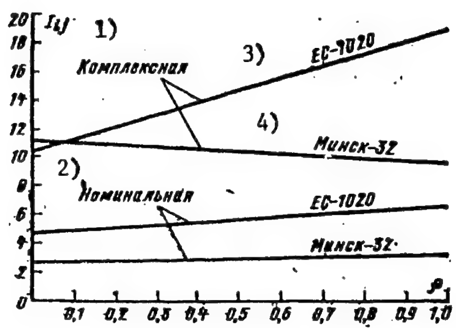


Figure 11-1. For Calculation of Productivity Indexes of Computers

Key:

1. Complex
2. Nominal
3. YeS-1020
4. "Minsk-32"

Comparative estimates of indexes of nominal productivity for these computers in solving scientific and economic problems are given in table 11-1.

Table 11-1.

Index of nominal productivity	Type of computer		
	"Minsk-22"	"Minsk-32"	YeS-1020
In solving scientific problems	1	2.6	4.6
In solving economic problems	1	3.7	11.3
Average	1	3.1	6.1

Calculation of complex productivity is rather complicated since many parameters of a computer and their influence on one another are not subject to an analytical description. Basically in P_k the influence of the capacity of the storage and

channel parameters are taken into account. Comparative estimates of indexes of complex productivity are presented in table 11-2 similarly to table 11-1.

Table 11-2.

Index of complex productivity	Type of computer		
	"Minsk-22"	"Minsk-32"	YeS-1020
In solving scientific problems	1	11.2	10.7
In solving economic problems	1	9.9	19.0
Average	1	10.1	15.9

Differences in the standard set of external storage units for the YeS-1020 computer for solving scientific and economic problems were taken into account in calculating the indexes in table 11-2. In particular, lowering of index P_k for the YeS-1020 computer in solving scientific problems as compared with a "Minsk-32" computer is caused by the small number of magnetic disk storages in the typical structure of a YeS-1020 computer.

Computer productivity indicators themselves cannot serve as criteria for their efficiency. Those computers which have a minimum cost/productivity ratio are economically more efficient. Comparative estimates of the cost of a unit of productivity for various computers are presented in table 11-3.

Table 11-3.

Cost of unit of productivity	Type of computer		
	"Minsk-22"	"Minsk-32"	YeS-1020
Nominal	1	0.44	0.35
Complex	1	0.13	0.2

The rise in cost of a P_k unit for the YeS-1020 computer as compared with a "Minsk-32" computer is caused by the high cost of external equipment for YeS-series machines. For the purpose of making fuller allowance for the efficiency of each unit included in the structure of a computer it is necessary to use the "cost-utilization" and disbalance factors which are discussed below.

CONTENTS

8831
CSO: 1863/24

HARDWARE

UDC 621.311.001.57

POSSIBILITY OF USING MIR-2 DIGITAL COMPUTER IN HYBRID COMPUTER SYSTEMS

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 81 (manuscript received 10 Sep 79) pp 101-103

[Article by Tamara Nikolayevna Malakhova, group leader, Siberian Scientific Research Institute of Power Engineering, Novosibirsk, and Yakov Mironovich Shlemenzon, senior engineer, Siberian Scientific Research Institute of Power Engineering, Novosibirsk]

[Text] To develop the majority of algorithms for digital control of complex electric power industrial facilities, it is most expedient to make use of hybrid (analog-digital) computer systems (GVS). In the process, the problem is solved the natural way: The object of control is modeled on an analog computer or physical analog, and the apparatus for data acquisition and transmission, channels for remote information and indication, and the digital complex proper are modeled on an electronic computer.

The researcher must have a tool that permits easy debugging and correction of programs, observing the graphic process of solving the problem and operating in the interactive mode. To this end, it is convenient to include the MIR-2 computer in the hybrid computer system. This machine has a convenient language; it allows easy detection and correction of errors in a program; it has a display that permits following the course of solving the problem both in digital form and in the form of a graph. The cost of a MIR-2 computer is not high.

Methods of including the MIR-2 computer in the hybrid computer system are based on extending the external language of the computer through introduction of microprogram control of a push broach that provides for execution of the additional statements needed for operation of the hybrid computer system. Using this method, the capability of the machine is fully utilized, the speed of the computer is not reduced practically and it is possible to construct a hybrid computer system with rather good technical characteristics [1]. However, this requires thorough knowledge of the computer design, high skill of thk researcher and a large amount of time for development and modernization.

For many problems, a more convenient method of including the MIR-2 computer in a hybrid computer system is to extend the capabilities of the computer by introducing additional numbers for the information input/output devices; this requires no modernization of the computer itself and allows exchanging analog information at the expense of insignificant complication of the scheme for conversion devices.

This method was implemented at the SIBNIIE [Siberian Scientific Research Institute of Power Engineering] on a hybrid computer complex (GVK) that includes the MN-17M and MN-18M analog machines, the UP-6 converter and an electrodynamic model [2]. In the process, the additional external devices were assigned the following numbers:

"VYVOD" 31 -- ATsP [analog-to-digital converter] channel address output

"VYVOD" 35 -- TsAP [digital-to-analog converter] channel address output

"VYVOD" 36 -- output of information from the digital computer to the digital-to-analog converter

"VVOD" 37 -- input of information from the analog-to-digital converter to the digital computer.

Since it is possible to input digital information into the MIR-2 computer only in the form of a word-formula clause, a device was developed in the hybrid computer complex that forms from thk digital information a clause of the form:

PUST U[T] = -6123, + 2024, ... KON ◇.

where T is the number of channels for analog-to-digital conversion.

Analog information in the UP-6 is converted into 12-place code with the point fixed after the sign bit; in thk process, the initial 12-place code is divided into triads and a sequence of binary-octal numbers is input into the computer.

The structure of the commands for exchange has the following form:

a) for input of multichannel analog information into the computer:

VYV 31 [AB]; VVO 37;

b) for output of multichannel digital information from the digital computer:

VYV 35 [AB]; VYV 3N S, D, Ye;

where A and B are the numbers of the initial and final channels for conversion; S, D and Ye is information to be output to the digital-to-analog converters.

The block diagram of the device for forming the clause (UFP) for the hybrid computer system is shown in thk figure.

Organization of information exchange in thk hybrid computer system occurs the following way. The channel connector (UVK) for the MIR-2 computer shapes the channel connection signal, by which the numbers of the initial and final channels for conversion are put into the UP-6 converter address registers. Then, as a function of the form of the next statement (VVOD 37 or VYV 3N), either analog information is input through the UFP [clause forming device], or information processed by the computer is output through the digital-to-analog converter.

Basic Technical Data for the Hybrid Computer System:

Number of channels for analog-to-digital conversion	24
Number of channels for digital-to-analog conversion	24
Number of amplifiers for sampling and storage in channels for analog-to-digital conversion	15

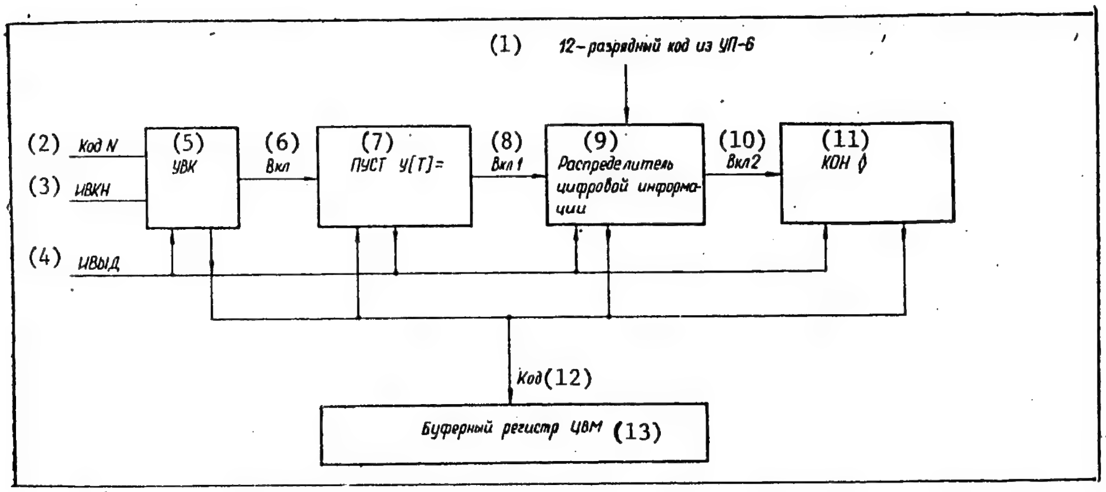


Fig. Clause forming device (UFP) for the GVS [hybrid computer system]

Key:
1. 12-place code from UP-6 converter
2. code N
3. IVKN
4. IVYD
5. UVK [channel connector]
6. connection signal
7. PUST U[T] =
8. connection 1
9. distributor of digital information
10. connection 2
11. KON
12. code
13. digital computer buffer register

[Continuation of] Basic Technical Data for the Hybrid Computer System:

Number of amplifiers for sampling and storage in channels for digital-to-analog conversion	24
Range of variation of analog signals, V	± 50
Number of discrete signals, input into hybrid computer system	512
Number of one-place commands, output from hybrid computer system	256

The hybrid computer system code is 12-place, parallel, normal and inverse with the point fixed after the sign bit. Let us note that the hybrid computer system is connected to the electrodynamic model by 20 independent channels; it allows upon command from the computer connecting to the measuring bus any block of racks of analog machines, effecting PUSK [start], OSTANOV [stop] and ISKHODNOYE POLOZHENIYE [initial position] of the computer, performing the function of a time delay element within the range of 0.1 to 3700 ms, synchronizing the operation of the digital and analog parts and varying the place-length of transmission of information from the computer to the digital-to-analog converter while simulating remote control channels. Additional apparatus (four printed circuit boards) is placed in the free spaces in the UP-6 rack.

The main shortcoming of this hybrid computer system is its low speed caused by the considerable time for operation of the subroutine for syntax checking prior to each input of analog information into the computer. However, the time relationships in the hybrid computer system implemented allow handling the majority of problems of digital control of complex power industrial facilities with a considerable number of switching apparatus and extended channels of communication. In particular, this hybrid computer system has made it possible to work out the main algorithms for data acquisition and processing for the ASU TP [process control system] for one of the actual 500 kV substations, perform analysis of normal conditions and static and dynamic stability with simulation of various devices and actions of supervisory personnel and modeling of the sequence of actions of various devices for relay protection and emergency relay controls, and to compute information indicators of the quality of operation of various units in the modeled system [3].

BIBLIOGRAPHY

1. Pukhov, G. Ye.; Samoylov, V. D.; and Aristov, V. V., "Analogo-tsifrovyye ustroystva modelirovaniya" [Analog-to-Digital Devices for Modeling], Kiev, Tekhnika, 1974, 324 pages.

2. Shlemenzon, Ya. M. and Kostrzhitskiy, V. D., "Hybrid Computer System Based on MIR-2 Digital Computer with Microprogram Control," in "Gibridnyye vychislitel'nyye mashiny i kompleksy" [Hybrid Computers and Complexes], Kiev, 1976, pp 115-116.

3. Bushuyev, V. V., "Hybrid Modeling of Electric Power Industrial Systems," ELEKTRONNOYE MODELIROVANIYE, No 6, 1980, pp 72-78.

8545
CSO: 1863/10

UDC 681.518.54

PRINCIPLES OF HARDWARE DIAGNOSTICS

Moscow OSNOVY TEKHNICHESKOY DIAGNOSTIKI in Russian 1981 (signed to press 11 Mar 81) pp 4-6, 317-319

[Annotation, preface and table of contents from book "Principles of Hardware Diagnostics: Optimization of Diagnostic Algorithms, Hardware" by Pavel Pavlovich Parkhomenko, Yegor Sergeyevich Sogomonyan, Yu. L. Tomfel'd and A. A. Odintsov, edited by P. P. Parkhomenko; editorial staff: academician V. A. Trapeznikov, academician A. A. Voronov, doctors of engineering science A. G. Mamikonov and O. I. Aven, and candidate of engineering science D. M. Berkovich; from the series on "Application of Computers to Research and Management of Production", Energoizdat, 9000 copies, 320 pages]

[Text] Methods are presented for obtaining optimal and optimized unconditional diagnostic algorithms; principles of the theory of questionnaires are presented. General structural approach is considered for the problem of building systems and means of functional diagnosing of the technical order of continuous and discrete objects.

Authors discuss principles of building external general purpose diagnostic means and the basic assemblies and units for them.

For engineers and scientific associates engaged in design, development and research of systems for diagnosing the technical order of complex objects.

Preface

This book is a continuation of the book published in 1976 [1-5] in which the principles of organization of diagnostic support of complex objects were presented, the main problems of technical diagnostics were discussed, and methods of selecting check points in building systems for diagnosis of the technical order of continuous objects by the tolerance method and methods of building tests for discrete combination and sequential devices were described.

In this book, we present the problems of optimization of diagnostic algorithms, the principles of organization of functional diagnosing of the technical order of complex continuous and discrete objects, the methods of synthesis of built-in monitoring circuits for discrete devices, the questions of organization of self-diagnosis and self-repair of discrete objects, and the principles of building external means of test diagnosing.

It msu be noted that since the publication of the first book, terminology in the field of hardware diagnostics has changed somewhat. Thus, for example, while in [1-5], the process of determining the technical order of an object was called diagnosis, and the results of this process, the results of diagnosis, now in this book in accordance with GOST [State Standard] 20911-75, "Technical Diagnostics. Basic Terms and Definitions," the process of determining the technical order of an object is called diagnosing [diagnostirovaniye] and its results the diagnosis [diagnoz]. Accordingly, the terms "object or system of diagnosing" are used in place of "object or system of diagnosis" etc. Also, a fault [neispravnost'] as a physical phenomenon is called a defect [defekt] in accordance with the standard, and the term "fault" is used either as a name of a type of defect (for example, logic or constant fault) or in the sense of an object or component of it being out of order. The term "failure" [otkaz] is used in the generally accepted sense of an occurrence that consists in the disruption of the proper functioning of an object or a component of it, while the term "error" [oshibka] means the appearance, caused by a defect (fault), of incorrect values of signals at the outputs of an object or component of it. The term "algorithm" is used in the ordinary sense as a formal procedure, the execution of which leads to obtaining a solution to a certain problem. Algorithms that solve problems of determining the technical order of objects (diagnostic problems) are called diagnostic algorithms.

The authors wish to express their thanks to I. M. Sindeyev and V. N. Zakharov for their thorough review and editing of the manuscript. The authors are especially grateful to their colleagues, associates of the laboratory of technical diagnostics at the Institute of Control Problems, for their active participation in discussing the material presented in this book.

Chapters 1 and 2 were written by P. P. Parkhomenko, 3 to 7 by Ye. S. Sogomonyan, chapter 8 by Yu. L. Tomfel'd and section 8.3 by Yu. L. Tomfel'd together with A. A. Odintsov.

The Authors

Table of Contents — Page

8545
CSO: 1863/267

UDC 681.325

INTERFACE FOR PROGRAMMABLE INSTRUMENTS IN EXPERIMENT AUTOMATION SYSTEMS

Moscow INTERFEYS DLYA PROGRAMMIRUYEMYKH PRIBOROV V SISTEMAKH AVTOMATIZATSII EKSPERIMENTA in Russian 1981 (signed to press 19 Mar 81) pp 2-3, 5, 261-262

[Annotation, introduction and table of contents from the book by Nikolay Ivanovich Gorelikov, Aleksandr Nikolayevich Domaratskiy, Sergey Nikolayevich Domaratskiy, Vitaliy Alekseyevich Liskin, Nikolay Vasil'yevich Popenko and Leonid Semenovich Sitnikov, Izdatel'stvo "Nauka", 4250 copies, 263 pages]

[Text] The monograph is devoted to interfaces usable to assure orderly exchange of information between autonomous measuring instruments, peripherals and a computer. The interface IEC BUS is described, the basis of which is the Hewlett-Packard interface HP-IB for programmable measuring instruments. The interface functions and control sequences are described in a high-level language. Algorithms for information exchange between instruments over the main line IEC BUS are described. Examples of the practical realization of interface cards for various instruments, computers and peripherals are examined, as well as questions of the construction of measuring information systems based on the described interface.

Introduction

The intensification and increase in cost of scientific investigations are causing an urgent need for the creation of systems for the automation of experiments. The principal tasks of such systems are the automated collection, registration, processing and presentation of information, and when necessary the management of experiments, which the unification of measuring, registering and controlling instruments into a single complex. Very effective is the creation of systems of experiment automation based on the combination of instruments of a single standard information main line, which assures a modular structure of the system and use of the principle of program control. Recently considerable efforts have been undertaken in this direction and a number of standards have been proposed, the best known of which is the CAMAC standard. The CAMAC standard has found its widest application both in the automation of scientific investigations and in industry in cases where a need arises for the creation of automation systems, including computers and having a variable set of measuring transformers and sensors, devices for input and output of experimental information, measuring instruments and peripherals. However, experience in the use of the CAMAC standard accumulated by the present time permits also noting a number of limitations intrinsic to it, the most important of which is considerable hardware redundancy, at times unjustified in applications where maximum speed is not required. Very suitable for construction of automation systems with a limited set of measuring and control equipment is a

standard which would permit using in the system any instruments, including those issued by industry and at the moment not adapted to system application. Of course, no one interface is ideal and capable of satisfying identically well many, often mutually exclusive requirements. Still the Hewlett-Packard standard, which serves as the standard of the International Electrotechnical Commission (IEC) for the interface for programmable instruments, permits successfully solving most problems in the creation of measuring and control systems which have fairly broad functional possibilities. The IEC interface assures a combination of instruments into systems by simply connecting them to a single standard information main line. A great merit of this standard is the absence of limitations on instrument designs and assurance of possibities of connection with or without small modifications of practically any instruments produced by industry. The merits of the IEC interface are noticeably displayed in cases where the user or developer has a one-time problem in the creation of an experiment automation system (a test system) and there is neither time nor resources for special developments for that system. The IEC standard coupler permits in that case readily uniting in any combination measuring, registering and controlling instruments, computers and peripherals with one another into a desired system.

At the present time the IEC interface is widespread all over the world. Such well-known companies as Solartron, DEC, DGE and Honeywell include the standard IEC coupling in their articles. It can be asserted with confidence that the IEC interface will find wide distribution in our country as well. However, its distribution is made difficult by an absence of publications containing not only a detailed description of the standard but also examples of the practical realization of its hardware and software. The present book has the purpose of eliminating that gap if possible and contributing to the very rapid introduction of the IEC interface. The book is intended for a broad circle of specialists, primarily instrument-makers and developers of experiment automation systems and industrial testing systems. The book can also be useful to specialists in automated control systems, engineers and scientific workers directly engaged in experiment in very different regions of science (physics, chemistry, geophysics and oceanology, biology and medicine, etc) and students of the corresponding VUZ specialties. Figure B.1 presents a reading algorithm which will help in work with the book and will enable time to be saved in studying the IEC interface.

Work in the writing of the book was divided among the authors in the following manner: Chapter 1 was written by N. I. Gorelikov, A. N. Domaratskiy and L. S. Sitnikov, Chapter 2 by A. N. Domaratskiy, S. N. Domaratskiy and N. V. Popenko, Chapter 3 by V. A. Liskin and N. V. Popenko, Chapter 4 by A. N. Domaratskiy, S. N. Domaratskiy and L. S. Sitnikov, Chapter 5 by S. N. Domaratskiy, and Chapter 6 by N. I. Gorelikov, V. A. Liskin and N. V. Popenko. Chapters 1 and 4 were edited by N. V. Popenko.

The authors would like to thank A. G. Blagodarev, M. M. Borkovskiy, K. V. Grinberg, V. A. Merzlyak and P. V. Shcherbakov for participating in the development of a number of devices described in the monograph, and L. I. Lyubomudrov and G. I. Mesyatsev for a number of valuable comments on the arrangement of the material and the formulation of the work.

CONTENTS

2174
CSO: 1863/30

UDC 51:621.391

MACROCONVEYOR COMPUTATIONS OF FUNCTIONS ON DATA STRUCTURES

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 13-21

[Article by V. M. Glushkov, Yu. V. Kapitonova, A. A. Letichevskiy and S. P. Gorlach]

[Excerpts] The basic principles of macroconveyor organization of computations, a promising reserve for raising the productivity of multiprocessor computer systems, were finally formulated in 1978 and are reviewed in work [1]. The realization of macroconveyor computations in the structure of multiprocessor systems and their software make it necessary to develop a corresponding theory. Work [2] proposed an apparatus which permits a quantitative evaluation of the of macroconveyor organization of the work of a system of processors when computing the values of functions on data structures. This article presents all the essential constructions and proves the statements that comprise the content of this apparatus. In addition it reviews a number of examples that show the possibility of using macroconveyor organization to solve problems of computer mathematics.

In conclusion, we will note certain possibilities of improving evaluations of efficiency. In the first place, synchronous realization is by no means always the best way from the standpoint of efficiency. The time spent in a system for exchange among modules can be reduced if we consider so-called asynchronous realization where computations in some processes coincide in time with exchange processes in others. In the second place, in some problems a significant improvement in the evaluation of efficiency can be achieved by a more complex distribution of data among processors. Thus, in addition to the partitioning $Q_i(p, n)$ it is possible to consider the covering $P_i(p, n)$ of domain $Q(n)$, where $Q_i(p, n) \subseteq P_i(p, n)$. The sets $P_i(p, n)$ for different values of i may intersect, but

$$\bigcup_{i=1}^{p} P_i(p, n) = Q(n).$$

in this case processor M_i stores the values of all initial data structures (that is, those which have values before problem-solving is begun) at points of domain $P_i(p, n)$ and computes the values of all data structures at points of domain $Q(p, n)$. The covering of domain $Q(n)$ makes it possible to reduce the number of exchanges in the modules by precluding exchanges of raw data. For example, in the case of multiplication of matrixes (the first variation of partitioning),

all exchanges are indeed done with raw data. Therefore, if the covering is constructed so that there are corresponding to the quadratic block of partitions the block of quantity A from the same line as the partition block and the blocks of quantities B from the same column, then exchange among processes does not take place. In this case efficiency is equal to one.

In some problems it seems wise to construct the covering not only for raw data but also for certain other types of data structure. The duplication of computations that occurs in this case may be advantageous because it reduces the number of exchanges.

FOOTNOTES

1. V. M. Glushkov, Yu. V. Kapitonova, and A. A. Letichevskiy, "One Approach to Realization of Parallel Computations in Multiprocessor Computer Systems," in "Parallel'noye Programmirovaniye i Vychislitel'nyye Sistemy" [Parallel Programming and Computer Systems], Novosibirsk, 1980, 1, pp 23-25.

2. V. M. Glushkov, Yu. V. Kapitonova, and A. A. Letichevskiy, "The Efficiency of Parallel Computation with Limited Resources," DOKL. AN SSSR, 1980, 254 No 3, pp 527-530.

3. V. M. Glushkov, Yu. V. Kapitonova, and A. A. Letichevskiy, "Theory of Data Structures and Synchronous Parallel Computations," KIBERNETIKA, 1976, No 6, p 215.

4. V. M. Glushkov, Yu. V. Kapitonova, and A. A. Letichevskiy, "Toward a Theory of Designing the Hardware and Software of Multiprocessor Computers," KIBERNETIKA, 1978, No 6, pp 1-15.

5. Lamport, L., "The Parallel Execution of DO Loops," COMMUNICATIONS ACM, 1974, 17, No 2, pp 83-93.

11,176
CSO: 1863/16

UDC 681.31

COMPUTERS WITH STRUCTURAL REALIZATION OF HIGH-LEVEL LANGUAGES

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 73-81

[Article by V. M. Glushkov, S. D. Mikhnovskiy and Z. L. Rabinovich]

[Text] 1. Introduction

Structural realization of high-level languages (HLL's) in computers involves, in the first place, recording the working (directly run) programs in HLL's close to the programming language and, in the second place, interpretation of this language by structural means (hardware and microprograms). The first circumstance makes it possible to insure a correspondence that can be understood by the user between his program and the operations performed by the machine and provides a rather simple translation of this program into the working program. The second circumstance makes it possible to insure efficient interpretation of working programs even if they are recorded in a complex language. It is unnecessary to list the advantages which these circumstances give. They have already been treated in the literature [1, 4] and a discussion of the advisability of realizing HLL's is unnecessary, particularly because in practice this advisability for small machines was first confirmed by building the domestic Mir computer (later, to some degree, by the Nairi also), while for large machines it was confirmed by development of the Burroughs family of computers in the United States and the domestic El'brus multiprocessor complex [5]. However, the idea of building large machines that realize HLL's was being worked out in the USSR as early as the 1960's. One of the first (if not the first) attempts in this area was development of the design of the Ukraina computer, based on original structures and principles of realizing HLL's, the control unit, and the operations unit [6, 5 and 11]. This machine was not produced, simply because of the lack of basic technological elements, which is a very important factor for efficient construction of machines that realize HLL's. At the present time the material-technical conditions of building such machines (on a contemporary scale even small machines would tend to be classified this way) have been fully prepared, and this applies also to computers that realize significantly elaborate HLL's and families of them. These conditions were used in building new computers, and not one of the development projects of this sort has been able to get by without solving the problem of realizing HLL's in one degree or another. Because interest in available experience in this respect has increased significantly, the authors consider it useful to publish an article that illuminates the basic concepts which were envisioned

by the design of the Ukraina machine and relate to its input language, internal language, interpretation system, and system for dynamic distribution of memory.

The article as a whole is historical, so it contains terminology which the authors used during the process of these development projects.

2. Input Language

The machine's input language was based on the algorithmic language ALGOL-60, supplemented by the following:

1. Means to simplify the programming of nonarithmetic problems (processing letter data in economic problems, formal conversions, and the like);

2. Means for more efficient use of internal machine capacities (performance of by-bit operations, filling memory densely using data units shorter than the length of a machine word, and so on);

3. Means for performing computations with enlarged bit configurations.

The input language of the machine differs from ALGOL-60 in that it has the following additional features:

1. Line quantities (of the STRING type) and line expressions;

2. Code quantities (of the CODE type) and code expressions;

3. Means for access to particular parts or fields of values of variables of the STRING and CODE types;

4. Means for assigning computations with enlarged bit configurations.

3. Internal Language

The internal language of the Ukraina machine is a multipurpose system of machine statements that is expected to provide the following:

1. Internal representation of initial programs written in a high-level algorithmic language, keeping the simple mutually unique correspondence between elements of the two forms of writing the programs (external and internal);

2. Efficient realization (interpretation) of statements by machine hardware and by the microoperations realized in it;

3. The possibility of translating initial programs into the machine's internal language without preserving the above-mentioned simple correspondence between elements of the external and internal records of the program, but using program optimization procedures in the machine's internal language;

4. The possibility of running programs at the level of the micro-operations of individual, rarely used algorithms for interpreting the machine's internal language;

5. Conventional, traditional control over the machine's peripheral units.

The program in internal language is recorded in the form of a sequence of elementary operations of the type

<Operator> :: = <Sign of Operation>
<Sign of Operation> <Index of Operand>
<Sign of Operation> <List of Operands>

In the general case the content of the action assigned by the sign of the operation depends on context. One or several operands may be assigned directly in the notation of the statement. One operand may be assigned by an index. The remaining operands, if there are such, are determined by the interpretation system.

A distinction is made between the problem-oriented and system-oriented parts of internal language based on the correspondence of the statements of internal language to elements of the input language.

The sign of the operation in each problem-oriented statement corresponds to a certain separator (not necessarily to the sign of an operation or a service word of the input language). In the system-oriented part of the internal language operation signs designate actions performed on elements of the hardware. The statements of the problem-oriented part of internal language are interpreted directly by machine hardware, and by procedures compiled in the system-oriented statements of this language. In addition, the statements of the system-oriented language are used to describe the machine's supervisory programs.

The internal language insures the possibility of keeping the block structure of the initial program, which is used as one of the mechanisms of dynamic distribution of memory.

The order of performance of statements of the internal language in the general case differs from the order of their arrangement in the text of the program.

A "sector" is a segment of the program within which the order of performance of statements is determined by a priority relationship adopted for them that is analagous to the priority relationship for arithmetic operations. Outside of a sector statements are executed in the order of their arrangement in the program notation.

The declaration of sectors is coordinated with the block structure of the program. A sector may be included in program blocks and include blocks in itself, without intersecting them, but it cannot include other sectors. The boundaries of sectors are marked off in the program notation by SECTOR BEGINNING and SECTOR END statements that switch the work mode of the interpretation system.

The internal language has an expanded composition of categories of objects which may serve as operands in the statements of this language. In addition to constants, simple variables, and arrays in the statements of the internal language, the operands may be arrays, markers, names of decision boxes, names of procedures, absolute addresses of points in the program, and addresses of memory cells.

The permissible types of values for constants, simple variables, and array elements and the possibilities of coordinating them in the internal language are the same as in the input language.

The construction of the problem-oriented part of the internal language is based on the principle of dividing the entire aggregate of actions to interpret programs written in the input language into groups of actions comparable to the particular separators of that language. As a result of this, the composition of internal language statements is very similar to the composition of separators of the input language and their classification reflects the semantics of the separators in the interpretation system adopted here.

When programs in the input language are being converted into machine notation, the delimiters of the input language are represented in the signs of the statements corresponding to them and the identifiers and markers that follow them are expressed in the indexers of the operands of these statements. Notations with service information that make the process of program interpretation easier are also included in the text of the program, in addition to constants, as operands of the statements of the internal language. This service information may be, for example, addresses that link blocks that are included in one another and the specifying markers contained in them.

The statements of the first group of the problem-oriented part of the internal language, which have an obvious relationship to the elements of the input language, are the following:

1. Statements that fix the boundaries of memory loading (BEGIN, END);

2. Arithmetic statements (use a +, -,

3. Logical statements ($\wedge$, $\vee$, $\supset$, $\equiv$, $\neg$);

4. Statements of relationship ($<$, $\leq$, $=$, $\neq$, $>$, $\geq$);

5. Switch statements (GO TO, SWITCH, ELSE, and COMMENT skipping of the segments of the program segments corresponding to these constructions, and the conditional switch statement THEN);

6. The "marker" statement, which corresponds to the marker of the initial program and corrects memory distribution in connection with the execution of switches;

7. Description statements. This group of statements, which insures memory redundancy for the values of variables and arrays includes variable description statements (REAL, INTEGER, CODE, COMPLEX, BOOLEAN, SYMBOL, and STRING and statements with the character "," in the list of variables), array description statements (ARRAY, statement with the character "," in the list of array identifiers, the statement [for the beginning of a segment, the statement : for a boundary pair, and the statement] for the end of a segment), the statement ' to indicate line length, and the statement of statistical distribution of memory OWN);

8. Statements that indicate an element of the array ([, ",",]);

9. The assignment statement :=;

10. Cycle statements (FOR, STEP, UNTIL, WHILE, DO, and the statement END which closes the body of the cycle);

11. Procedure statements (PROCEDURE, statements of formal and factual parameters («,»), the statement; for entry to the body of a procedure, and the statement END which closes the body of the procedure).

Including these statements in the internal machine language makes it possible to preserve structural similarity between the text of the program written in the input language and the notation of this program in machine form.

The second group of problem-oriented statements of the input language comprises the statements which are expected to insure efficient realization of a broad range of computing and noncomputing procedures. They do not have such an obvious and direct relationship with elements of the input language. Some of these statements are very close in meaning or actually coincide with statements of the first group. The purpose of including this group of statements in the internal machine language is to insure the possibility that it can carry on the computing process not only on the level of practically direct interpretation of the input language received, but also use optimizing translation of the initial program and other high-level input languages.

The statements of this group are the following:

1. The statements of addition, subtraction, multiplication, and division, which differ by concluding operations to normalize and round the results;

2. Special statements to divide by numbers that define the structure of the machine word (48 and 6);

3. The statements of addition and subtraction of number moduli and to change their signs;

4. Statements of transformation of the type of values of quantities and the form of their machine representation;

5. Statements of logical shifts (cyclical and linear) in the notations of operands of the types CODE and STRING;

6. Statements of an arithmetic shift (linear and cyclical) in the internal machine representation of operands of the types REAL, INTEGER with a fixed decimal point;

7. Statements to count the total number and determine the place of assigned symbols 0 or 1 in operands of type CODE;

8. Statements for access (read/write) to parts (fields) of operands of types CODE and STRING;

9. Statements for by-character line processing (statements to replace and compare characters, to assemble and sort lines by mask);

10. Statements of unconditional switch with operation of the next return and/or authorizing or prohibiting an interrupt;

11. Statements of conditional switch according to the sign or zero value of the result.

Three groups of statements are identified in the system-oriented part of the internal language: (1) Microstatements for programming elementary operations on registers and control triggers of the interpretation system; (2) Statements of the system for dynamic distribution of memory; (3) Systems statements of the macroinstruction type designed to control equipment that is external to the processor.

The microstatements are divided as follows according to operations performed:

1. Microstatements for exchange of codes between any assigned memory cell and the primary full- bit position registers of the central control unit;

2. Microstatements executed on codes in registers of the central control unit (writing a zero code, adding or subtracting units, adding or subtracting codes into registers, writing an assigned code in the registers);

3. Microstatements that set the state of control triggers;

4. Logical microstatements (statements of logical operations on the states of control triggers and the state of the so-called "operations" trigger which stores the result of the preceding logical operation, statements of logical operations on results of comparing the content of the assigned register with an assigned code as a boolean variable and the state of the "operations" trigger);

5. Microstatements of program branching (conditional switch based on the state of the "operations" trigger with possible preparation to return to the next microstatement).

The group of statements of the system for dynamic distribution of memory was specially introduced to simplify program realization of the algorithms of this system, which are used somewhat infrequently.

4. Principles of Interpretation of the Statements of the Internal Language

The most important aspects of the organization of interpretation of the internal language in the Ukraina machine are: dynamic control of the order of execution of program statements; determination of the semantics of statements depending on the context of the program by forming and registering in the machine a certain code called the interpretation mode; broad use of non-addressed storage of intermediate results and access to them as operands of the statements being interpreted; use of an integrated system of dynamic distribution of stepped memory (non-addressed storage of integrated results is one of the elements of the general system of dynamic distribution of memory).

In order to simplify the form of fixing the correspondence between program notations in the input and internal languages, the separators of the input language are also indicated as the signs of operations in statements of the internal language.

The separators of the input language by themselves do not insure unambiguous identification of the statements because their meaning depends on context. In a strict sense, the statement in internal language is defined by the sign of the operation and the mode established by the interpretation system at the moment that this statement is executed.

The mode of execution of statements of the internal language is represented by a control code formed in a dynamic complementary manner to the code of the sign of the operation. As a control code the mode has multiple components. One component, as pointed out, determines the place of the statement in the context of the program, while the others fix the conditions of execution of the current statement in connection with the mechanism operating in the system for scanning the text of the program within the limits of the segments declared by the sectors.

Within the limits of a sector the order of execution of statements is determined by a seniority relationship established for them. A reciprocating circuit operates for scanning the text of the program, and in connection with this three types of execution of statements are distinguished: direct, preparatory, and inverse.

Direct execution of a statement is execution of the statement from start to finish in its full volume. Where other conditions which will be mentioned below are observed, direct execution of a statement is possible if the sign of the operation in it is not junior to the sign of the operation of the following statement. Thus, in a certain fragment of the program...*B**C***D..., direct execution of the statement *B can take place if the operation * is not junior to the operation **. With direct execution of the statement *B where * is the sign of a certain two-place operation, the left operand (we will designate it A) at the moment of execution of operation * is stored in a certain register of the machine as the result of the preceding operation, while the right operand B is determined by the index corresponding to it in the notation of this statement. The result of the operation is stored in the intermediate result register.

If the operation * is junior to the operation **, preparatory execution of statement *B takes place. In the process of preparatory execution of the statement the current intermediate result is copied from the register into main memory, the sign of the operation (in this case *) is stored as an unexecuted sign, and the value of the operand (in the given case the operand B) is selected from memory and put in the intermediate result register. After this the possibility of direct execution of the following statement **C is tested.

After direct execution of statement **C, just as after direct execution of *B if it has taken place, the possibility of execution of the last sign of an operation that has not been executed and is stored in the interpretation system is tested. This kind of execution of a statement not executed earlier is called inverse.

With inverse execution operation * is performed on operand A as the left operand already stored in main memory and extracted without special indication of the address and on the result of execution of statements after **C as the right operand stored in the intermediate result register.

Inverse execution of an operation is possible if its level of seniority is not lower than the level of seniority of the operation in the next program statement. After inverse execution of the last unexecuted statement an attempt is made to execute the next unexecuted statement, and only if it cannot be executed or there is no such statement is the possibility of executing the next statement in the program notation tested.

5. The System of Dynamic Distribution of Memory

The memory units of the machine are classified in two groups according to type and manner of use: the units that make up internal memory, and the external memory units.

The internal memory, which for the most part is distributed dynamically, consists of the main ferrite memory unit, the fast ferrite memory unit, and a magnetic drum store. Parts of the fast and main memory units may, if desired, be excluded from the dynamic distribution system or, on the other hand, be transferred under this system. The magnetic drum store and dynamically distributed part of the main ferrite memory unit are covered by the general mathematical page numbering of the cells. The physical memory represented by this memory unit is broken down into pages which contain 1,024 cells apiece, while the range of mathematical addresses used is subdivided into intervals or segments commensurate with them.

The number of segments of mathematical addresses is selected to be larger than the number of pages of physical memory, so that the space of the mathematical addresses operates as a distinct distributed machine resource, the mathematical memory.

The pages of physical memory are dynamically assigned to segments of mathematical memory. A segment of mathematical addresses receives a certain assigned page only after actual access to it. In order to reduce the length of the mathematical addresses put in the indexes of operands of the internal language, the mathematical memory is subdivided into 28 blocks and certain (given below) agreements relative to the placement of information in these blocks are adopted. This makes it possible for the program to use chiefly "truncated" 16-bit addresses, relating them to a block of mathematical memory that is known at each particular moment.

The system of dynamic distributed memory in the Ukraina machine is based on:

- Generalized magazine organization of memory for storage of intermediate results registered by the interpretation system;

- Explicit representation in the machine program of the block structure and description of variables of the initial program as a mechanism of occupying and freeing main memory;

- Uniform mathematical addressing of the main and dynamically distributed parts of internal memory (so-called pseudomain memory) on magnetic drums;

- Page organization of memory distributed not only on the primary main memory, the ferrite memory unit, and the pseudomain memory, but also the fast main memory based on the small-volume ferrite memory unit;

- Distribution between the problems being solved of the mathematical addresses and the real, physical memory, as two independent resources.

We give below a description of the subsystems particular aspects of the organization of the general system of dynamic distributed memory.

Distribution of memory as a whole is achieved by distribution of memory (mathematical and physical) among problems being solved and mathematical memory within the framework of each individual problem and distribution of information among steps of physical memory.

5.1. Distribution of Memory Among Problems Being Solved

The distribution of memory among problems being solved is based on the use of an expanded range of mathematical numbers of cells and dynamic assignment of pages of physical memory to them. The mathematical memory is distributed among problems a priori (statically) based on maximum need for them. A certain range of mathematical addresses is assigned to each problem. The physical memory is distributed among problems dynamically because its pages are assigned to segments of mathematical memory only in connection with their actual use.

5.2. Distribution of Mathematical Memory of an Individual Problem

The range of mathematical addresses allocated for a problem is distributed by the actions of: (1) The "reversive" circuit for occupying and freeing addresses; (2) The circuit for addressing variables and arrays; (3) The system for addressing links between statements of the program.

The occupying and freeing of mathematical addresses in the range allocated to each particular problem is done by a so-called reversive circuit which is a generalized form of the well-known magazine circuit.

Mathematical addresses can only be occupied and freed sequentially from the two ends of the assigned range so that composite addresses always form a single interval with boundaries m and M. At each step of distribution of mathematical memory either one address or a group of consecutive addresses can be occupied (freed) from either end of this interval. In the interpretation system the possibility of access is provided not only for the last occupied (from either side) mathematical address, but also for any address counted from values of m and M known to it.

The reversive scheme of dynamic distribution of mathematical memory is accomplished by a selected order of registering and using intermediate results that arise during scanning of the program text, storing the block structure of the initial program in the machine program, and distributing the circuit for occupying and freeing mathematical addresses when feeding to the block and outputting from it to all the language constructions put into one another, including the recursively requested procedures.

The mathematical addresses may be occupied either for quantities put into the computing process explicity by means of special descriptions or by quantities which do not have such descriptions, as well as by intermediate results and service information envisioned by the interpretation system adopted.

The storage of the current mode and setting of the new mode when executing statements that correspond to the beginning of such segments of the program as the list of boundary pairs or list of formal parameters and the restoration of a former mode when executing statements that correspond to the end of these segments may serve as examples of the distribution of the block principle for occupying and freeing mathematical addresses.

In order to increase the efficiency of dynamic distribution of information among steps of memory, the occupying of mathematical addresses for various categories of quantities is regulated by additional conditions that facilitate arrangement of information in different pages of memory depending on the frequency of its use.

Addresses for the values of simple variables, quantities that are not described, the service information of the interpretation system, and the intermediate results occupy the free mathematical addresses (from the address m) from the left edge of the range. The addresses for elements of arrays occupy addresses from the opposite, right edge of the range (from the address M), but the service information necessary for access to these arrays (minimum and maximum values of indexes, lengths of arrays for each measurement, and addresses of the first elements of arrays) occupy addresses in the left part of the range in the form of so-called "reference words."

The addresses for the user's own variables and arrays are occupied based on the results of a special scan of the program before it begins to be run.

The program is arranged in a separate block of mathematical memory from the data.

The addressing of variables and arrays in the program notation. With the scheme described above for occupying the mathematical addresses, for each simple variable and array of the basic program there is a completely definite mathematical address that does not depend on the course of execution of this program and the actual dimensions of the arrays. For the simple variable this is the address of its value, while for an array it is the address of its "reference word." In order to preserve this status with respect to procedures also, including procedures executed recursively, the indexes of the operands for simple variables and arrays indicate not absolute but relative mathematical addresses counted from the address m at the moment of initiation of execution of the procedure in which these quantities are described. This value m' is registered by the interpretation system in special index registers to which access is made in conformity with the so-called static rank of the procedure.

The static rank of a procedure is a parameter that determines the static inclusion of procedure descriptions in one another. The program is viewed as a 0-rank procedure. The procedures described directly in the program have a static rank equal to one, and so on.

At each moment in time index register i contains address m' which corresponds to a procedure with static rank i currently being executed (including a procedure being executed recursively).

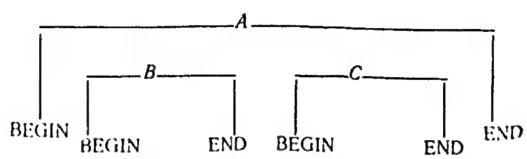
The address linkages among statements of the program. There are three types of address linkages among elements of a machine program: (1) Between markers in the GO TO statements and descriptions of switches, and also among specifying markers which mark statements that accept control; (2) Between the initial and terminal points of segments of programs that are omitted for some reason or other in the process of interpretation (for example, segments corresponding to procedure descriptions and parts of a conditional statement); (3) Among blocks included in one another.

The need for linkages of the first type is dictated by the semantics of the corresponding statements. Linkages of the second type are included in the program to skip sectors of the program that are unnecessary in particular concrete situations. Linkages of the third type are used to solve the problem of rejecting preoccupied mathematical addresses in connection with departure from the block either through its end or based on a switch statement.

At each moment in time the registers of the interpretation system register the address m' and M' which occurred at the moment the program being executed at this moment was fed to this block. Upon departure from the block the values of the addresses are restored.

If entry to the next block is made before departure from the particular block, following the BEGIN statement the registered values of the addresses m' and M' are copied from the registers into memory as service information that is stored by the interpretation system. These addresses are written according to the current mathematical address m as part of the machine word which acts as the operand of the BEGIN statement.

Thus, in any system of blocks, for example, the set of blocks shown below



the addresses m' and M', which must be set for block A upon departure from block B or C, are taken from the word, the operand of the BEGIN statement of block A. To support this operation the END statements of blocks B and C indicate the address of the BEGIN statement for block A, which directly encompasses them. To solve this problem when departing from the block on the GO TO statement, the address of the BEGIN statement of each block is also indicated in the statements which correspond to the specifying markers localized immediately in this block.

5.3. Distribution of Information Among Steps of Memory

The system for dynamic page distribution of information encompasses three steps of memory: the basic ferrite memory unit, the magnetic drum store, and the fast ferrite memory unit. Taken together they make up the machine's internal memory.

The page structure of memory and correspondence table. The information arranged in memory is broken down simultaneously into large and small segments in such a way that each small segment belongs entirely to a certain large one and each large segment consists of the same whole number of small segments. The complete number of a small segment is made up of its number in the large segment and the number of this large segment.

Each large segment of information must be located entirely on a certain page of either the basic memory unit or the magnetic drum store. Parts, small segments of large segments located in the basic memory, may be copied into the fast memory unit; in this case the places of the small segments in the basic memory unit are preserved. A constraint operates when segments are exchanged between the primary memory and the magnetic drum store: a large segment cannot be discarded from the basic memory and magnetic drum store if even one of its small segments is located in the fast memory unit.

Thus, the exchange of information among steps of memory, between the magnetic drum store and primary memory and between primary memory and fast memory, is done in segments of different size. It was necessary to use this kind of mixed page organization in the system for dynamic distribution of memory because of the significant difference in volume between the basic memory and the fast memory unit.

A two-step registration of the correspondence between the numbers of the information segments and the numbers of the pages in memory they occupy occurs. The first step is represented by the basic correspondence table, which covers all information segments and is stored in the main memory. The second step is the so-called active correspondence table, which is realized on trigger registers. A "line" in the active correspondence table is represented by a pair of registers that register the number of the information segment and the number of the page in memory that it occupies.

The active correspondence table is an abbreviated active step of the correspondence table in the sense that the number of lines it has is less than the number of pages in the main memory. It provides fast access to the physical addresses of only a limited number of the most actively used segments of information. There is a dynamic allocation of entries from the basic correspondence table to the active correspondence table with replacement of those that are not used so actively.

Each reference to the active correspondence table involves a reorganization of the lines contained in it, and as a result they are ordered by frequency of use. If the required information is located in a segment whose notation is located on line i ($i \neq 0$) of the table, this line changes place with lines ($i-1$). If the active correspondence table does not have the required segment the lines are shifted linearly in the direction of their higher-order numbers, and as a result the lines with the highest-order numbers are pushed out of the table and the entry with the required segment, located in the main corresponding table, is added on the first line of the active correspondence table.

The use of the abbreviated active correspondence table makes it possible to reduce hardware expenditures for its realization when using large main memory volumes and provides the possibility, if necessary, of building it up later. The fact that arrays of data are localized in mathematical memory separately from the more actively used information promotes efficient use of the abbreviated active correspondence table and also of the fast memory unit.

The active correspondence table provides access both to the large segments located in main memory and to parts of them, to the small segments put into the fast memory unit.

Requesting segments by groups. To accelerate the process of solving problems that deal with large arrays of data by reducing relative time expenditures in them to locate the necessary page in magnetic drum storage, segments of arrays from drum storage and main memory are requested in the system of dynamic distribution of memory by groups, not individually. In this the system takes advantage of the fact that the processing of actually used arrays of data with the selected size of large pages basically results in a cyclical sorting of segments within the limits of each particular array, that is, reference to the array consisting of segments K_0, K_1, ..., K_j ..., K_{p-1}, following the reference to segment K_j relates either to this very segment or to segment $K_{(j+h) \bmod p}$, where parameter $h = \pm 1$ determines the direction of sorting. The essential feature of a group request of assigned number n segments in connection with reference to segment K_j which is located in the magnetic drum store is that segments K_j, $K_{(j+h) \bmod p}$, $K_{(j+2h) \bmod p}$, ... are put in main memory, if not already there.

The direction of sorting of array segments is not known to the interpretation system; it is set before each new group of segments is called based on the results of a comparison of the number of segments from the series $K_{(j+h) \bmod p}$, $K_{(j+2h) \bmod p}$, ..., $K_{(j+nh) \bmod p}$; $k_{(j-h) \bmod p}$, $K_{(j-2h) \bmod p}$, ..., $K_{(j-nh) \bmod p}$, which are located in main memory at the given moment.

Each group request for segments is accompanied by actions to correct the size n of the requested group so that the time interval between two successive exchanges is maximal. The size of the group at each step of regulation changes by ±1 only after a definite number of consecutive instructions that such a change is useful. To limit the sphere of action of the mechanism for requesting segments by groups, in those cases where the assumption of cyclical sorting of segments is not correct in the specific array, a mechanism operates in the system to dynamically reduce the size of the requested group in those cases where the exchange between the main memory and magnetic drum storage involves segments which, having been requested in the main memory, return to the magnetic drum store without being used in the computing process.

Criteria for replacement of information segments in the operational steps of memory. The criteria for replacement of information in the operational steps of memory, the main memory and fast memory, rely on a set of possibilities provided by the use of a high-level language and its interpretation system as

well as features of the realization of the particular system of dynamically distributed memory:

- There is the possibility of considering the type of information contained in the segments and dividing them into operational segments, program segments, and segments of arrays;

- the presence in the program of special statements for access to elements of an array makes it possible to separate this group of references from references to other types of information;

- The use of an abbreviated active correspondence table allows the ranking of segments in the main memory according to the frequency of reference to them.

Operational segments, program segments, and segments of arrays form as the result of the system adopted for occupying mathematical addresses in the machine. Notes on the type of segment are put in the general correspondence table and are always accessible to the interpretation system.

Beginning from the fact that the average frequency of reference to the operational segments and program segments during the period of their active use in the general case is an order greater than the frequency of reference to the segments of live (consisting of more than one segment) arrays, the fact that such segments are pushed out of the active correspondence table is interpreted as an indication of the possibility of pushing them out of the main memory. The fast memory holds only small segments from operational segments contained in the active correspondence table. Pushing an operational segment out of the active correspondence table causes its small segments, if there are any, to be copied out of fast memory and main memory.

Because the sorting of elements of arrays in the computing process is regular in character and usually accompanied by cyclical sorting of their segments, the replacement of segments of arrays is accomplished on the basis of a prediction of the time of the next access to the segment. Segment replacement is done in conformity with an algorithm [12] that insures a minimum number of replacements with an appropriate modification for the case of segments requested by groups. In each step of the segment replacement process (if there are no operational or program segments pushed out of the active correspondence table), the segment removed from main memory is the one to which the next reference will come later than to the others.

The time until the next reference to a segment of the array is estimated by the dynamic distribution of memory system based on knowledge of the composition of the segments occupied by each of the arrays, the direction of their sorting, and the average time of processing a segment of each of these arrays, which is computed during the work process.

The use of such an algorithm for group replacement of segments with a stable computing process leads to stabilization of the section of pages of operational memory between arrays. In connection with this the replacement of segments of arrays in main memory is actually accomplished by performing two processes: (1) Internal replacement of segments in arrays with a fixed division of pages of main memory between them, when new segments of arrays requested from the magnetic drum store are placed in main memory in the place of the segments of those same arrays; (2) Replacement of segments leading to a change in the number of segments occupied by the arrays in main memory, which may occur in connection with placement of an operational or program segment in main memory or a change in the composition of arrays being processed, the number of segments requested at one time, or the time of their processing.

Internal replacement of segments of an array consists in the following: when n segments K_j, $K_{(j+h)\bmod p}$, ..., $K_{(j+(n-1)h)\bmod p}$ are requested from magnetic drum storage for main memory in connection with a reference to segment K_j, they are placed in main memory in the place of segments $K_{(j-h)\bmod p}$, ..., $K_{(j-nh)\bmod p}$. This kind of replacement is very simple to realize and under the given condition results in the minimum number of replacements of segments in main memory.

FOOTNOTES

1. V. M. Glushkov, A. A. Barabanov, L. A. Kalinichenko, S. D. Mikhnovskiy, and Z. L. Rabinovich, "Vychislitel'nyye Mashiny s Razvitymi Sistemami Interpretatsii" [Computers with Developed Interpretation Systems], Kiev, "Naukova Dumka", 1970, 258 pages.

2. V. M. Glushkov, and Z. L. Rabinovich, "Some Lines of Development of Algorithmic Structures of Computers," in "Kibernetiku -- na Sluzhbu Kommunizmu" [Put Cybernetics to Work for Communism], Moscow, 1967, pp 173-183.

3. V. M. Glushkov, S. B. Pogrebinskiy, Z. L. Ravinovich, and A. A. Stogniy, "Questions of the Development of Digital Computer Structures in Relation to Their Software Systems," KIBERNETIKA 1967, No 5, pp 15-28.

4. L. N. Korolev, "Struktura EVM i Ikh Matematicheskoye Obespecheniye" [Computer Structure and Software], Moscow, "Nauka", 1974, 254 pages.

5. Burtsev, V. S. "Principles of Construction of 'El'brus' Macroprocessor Computing Complexes," Moscow, 1977, 53 pages (Preprint ITM and VT AN SSSR, No 1).

6. V. M. Glushkov (topic director), "Scientific-Technical Report," SB. REF. NIR/IK AN USSR, Series 13, No 5, Inv. No A008718, Moscow, 1969.

7. V. M. Glushkov, G. M. Mikhaylov, S. D. Mikhnovskiy, Z. L. Rabinovich, A. A. Stogniy, V. Ye. Shamanskiy, K. I. Podkolzina, and V. P. Gladun, "Author's Certificate 219291 (USSR). Ditigal Computer," published in BYULLETEN' INFORMATSII 1968, No 18, 118 pages.

8. A. A. Barabanov, L. A. Kalinichenko, S. D. Mikhnovskiy, and Z. L. Rabinovich, "Author's Certificate 209640 (USSR). Control Unit for Digital Computers," published in BYULLETEN' INFORMATSII 1968, No 4, 121 pages.

9. S. D. Mikhnovskiy, "Author's Certificate 18139 (USSR). Technique of Dynamic Page Distribution of the Memory of a Digital Computer," published in BYULLETEN' INFORMATSII 1966, No 9, 106 pages.

10. S. D. Mikhnovskiy, "Author's Certificate 18562 (USSR). Technique of Dynamic Page Distribution of the Memory of a Digital Computer," published in BYULLETEN' INFORMATSII 1965, No 17, 93 pages.

11. S.D. Mikhnovskiy, "Author's Certificate 193578 (USSR). Technique of Dynamic Distribution of Information," published in BYULLETEN' INFORMATSII 1967, No 7, 54 pages.

12. S. D. Mikhnovskiy, and N. Z. Shor, "Estimate of the Minimum Number of Copying Operations in Dynamic Distribution of Page Memory," KIBERNETIKA 1965, No 5, pp 18-20.

11,176
CSO: 1863/16

UDC 519.6:681.3.06

SOME QUESTIONS OF PROBLEM-SOLVING ON COMPUTERS WITH PARALLEL ORGANIZATION OF COMPUTATIONS

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 82-88

[Article by V. M. Glushkov and I. N. Molchanov]

[Text] 1. Analysis of scientific-technical problems allows us to draw the conclusion that the problems which require the largest volume of computer resources are questions of nuclear and thermonuclear energy that come up in the process of designing reactors, performing strength calculations for the structures of atomic power plants and for aerodynamics and hydrodynamics, performing static and dynamic strength calculations of contemporary engineering projects as a whole, problems of the theory of climate and circulation of the atmosphere and ocean, calculations of the regimes of the USSR energy system, computing the transmission of vibrations in the Earth, and similar matters.

The nature of the problems that must be studied in the 1980's is changing. The need is arising to perform integrated investigations of objects and phenomena as a whole (for example, investigating the statically stressed state of a model of an entire aircraft), whereas in earlier times models of the elements and assemblies of the design or some particular aspect of the phemonenon were studied. The objects of study today are large in terms of geometric dimensions. For example, the hydrodynamic model of the circulation of the atmosphere and ocean is investigated for the entire surface of the earth. It is becoming very important to solve spatial problems. For example, in one of the stages of the problem of transmission of vibration within the earth it becomes necessary to solve a dynamic spatial problem of elasticity theory which cannot be reduced to a planar problem. More and more often the object of investigation is not merely the static state of the object under investigation, but also its dynamics (when designing atomic power plants, for example, strength is considered not only in a static state, but also under earthquake conditions, in the case of an airplane crashing directly into the power plant building, and so forth). Whereas in earlier times the models investigated were chiefly linearized models, it is now becoming necessary to study nonlinear models.

The problems of optimal control and operations research are changing qualitatively and quantitatively. The problems of optimal distribution of orders for production of pipe among enterprises of the USSR Ministry of Ferrous Metallurgy lead to the problem of linear programming with more than 500,000 variables and up to 50,000 constraints. The problems of future planning are reduced either to dynamic models of linear programming or mixed discrete-continuous models. In both cases the number of machine operations is at least an order greater than in problems of current planning.

The problems of siting production, arranging and standardizing technical equipment, and synthesizing networks (in communications, pipelines, computers, and elsewhere) involve multicriteria discrete-continuous models with a large number of variables. For example, for problems of synthesizing networks with concave functions of expenditures with one source and n users, the dimensionality of the evaluation problem of linear programming is roughly 10^3 variables and constraints. Significant computer resources are required to solve multiproduct and continual dynamic models that generalize known macromodels for a formalized description of complexly developing systems.

The problems of automated control systems that require large expenditures of machine time are focused around solving the problems of the OGAS [State Automated System for Collection and Processing of Information for Accounting, Planning, and Management in the National Economy], the RASU's [Republic Automated Control Systems], and calculations for operational control of the optimal loading of industrial equipment at enterprises of the republic ministries. Interesting problems from this point of view are the problems of operational accounting and monitoring of the production of output by the basic facilities of a combine, compiling drafts of quarterly and annual plans for truck shipments and determining optimal freight loads for a territorial motor vehicle transportation administration, predicting the route of travel of passengers with selection of a scheme for a system of air connections, compiling a central schedule of aircraft traffic, formulating a system for operational control of the shipment process in railroad transportation, and other similar questions.

2. With the introduction of simplifying hypotheses, scientific-technical problems are reduced to physical models which can be described by mathematical language. In this way mathematical models of applied problems arise. Most of the scientific-technical problems considered above amount to solving standard classes of mathematical problems that require significant volumes of arithmetic operations for solution.

There is already a need to solve systems of linear algebraic equations with tape matrixes (their order reaches 10^5 and the width of the tape is more than 1,000) with thinned symmetrical matrixes (their order reaches 1 million). Solving a generalized problem of characteristic (own) values with tape and symmetrical matrixes A and B, whose order reaches 10^4 while the width of the tape is 600-800 elements, requires computing several minimal characteristic values and the characteristic vectors belonging to them.

When nonlinear differential equations are solved by the finite elements technique, large systems of nonlinear algebraic equations occur, numbering into the thousands. When the finite differences technique is used several thousand such equations arise. Such large systems occur in the description of some applied problems (on the order of several thousand), but the number of related unknowns in each equation is not more than five percent.

With numerical integration of Cauchy problems it is necessary to solve systems of standard differential equations whose number reaches 10^4, or 10^3 in the case of rigid systems. When describing various applied problems, boundary problems arise for systems of standard differential equations whose number may reach several hundred.

In many applied problems it is necessary to find numerical solutions to three-dimensional elliptical, parabolic, and hyperbolic differential equations, both linear and nonlinear.

It is essential to find certain minimal characteristic values and the characteristic functions of differential operators corresponding to them, both of the second and of higher orders in partial derivatives.

The finite elements technique, the finite differences technique, the integral ratios technique, the straightline method, different variation techniques, and other methods may be used as means to discretize boundary problems for standard differential equations and equations in partial derivatives.

In some applied problems there occur linear integral Fredholm equations of the first and second types, the second type of convolution, and homogeneous integral Fredholm-type equations (the problem of characteristic values), Volterra linear integral equations of the first and second types, and systems of certain integral equations.

Numerical solutions to differential and integral equations must be obtained with a relative error of no more than three percent.

In problems of mathematical programming there is a need to solve problems of distribution-type linear programming with up to several hundreds of thousands of variables and several tens of thousands of constraints, to solve nonlinear problems of unconditional optimization with up to thousands of variables, to solve applied problems of discrete and discrete-continuous programming with up to thousands of variables, and to solve combinatorial optimization problems defined in reordering (relocation) space with up to several hundred generating elements.

In the course of solving applied problems, in addition to the above-enumerated classes of mathematical problems it is necessary to solve problems of statistical and probability analysis, numerical differentiation and integration, interpolation and approximation of functions, and computation of elementary and special functions.

3. Most scientific-technical problems are solved mainly in the main memory of computers, with only the subsidiary information necessary for the solutions stored in the second step of memory.

In a number of cases, however, the main memory is not adequate to solve the problems and during the solution process the second-step memory is used actively. It usually takes dozens of hours to calculate such problems.

There are also problems which are broken down at the level of the physical model into several sub-problems, and each of them uses the second-step memory during machine solution. The solution time for such a general problem may take hundreds of hours, and preparation of the raw data may take several worker-years.

The applied problems listed above are mainly problems of the third type.

The complexity of solving such problems lies in the significant volume of data processed, the trouble of preparing raw data, the development of special computing schemes that consider both the volume of data and the organization of second-step memory, and the inadequate precision of the machine result obtained (for some problems the best that can be done is to determine just the correct order of solution).

4. One means of accelerating the solution of scientific-technical problems that require large volumes of computation and a way to improve the accuracy of computations is parallel data processing.

In principle paralleling can be done at these levels: construction of physical models, objects, or processes of the phenomena under investigation, the solution method, algorithm, or program; exchange of data in the machine; inputting and outputting data.

Let us review a few of the existing ways of paralleling. Suppose it is necessary to make a study of the statically stressed state of a certain design element [1]. Considering the geometry of the object under investigation we mentally divide the element into several subelements which can be described fairly well by elasticity theory equations written in different systems of coordinates (see Figure 1 below). Additional boundary conditions must be formulated at the points where the mental division of the element is done. The eight subelements that result can be divided mentally into new ones or the calculation for the eight subelements can be organized simultaneously on computers. This is how paralleling is accomplished at the level of the construction of physical models.

The development of numerical methods of problem-solving on multiprocessor computers with parallel organization of computations requires new thinking and a new approach to the formulation of techniques. We will illustrate this with the theoretical example considered in [2].

Suppose that it is necessary to compute $y = x^n$. To solve this problem in the sequential computations mode several techniques can be worked out.

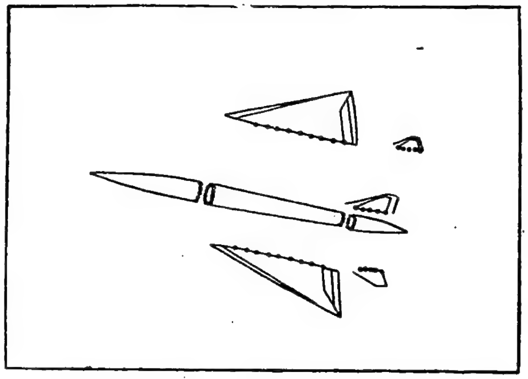


Figure 1.

The method of parallel computations of $y = x^n$ can be based on the correlation

$$\sum_{j=1}^{n} \frac{r_j}{x - r_j} = \frac{n}{x^n - 1},$$

where r_j is roots of the equation $x^n = 1$. $S_i = \frac{r_i}{n}$, $i = 1, 2, \ldots, n$ is calculated. Then for n processors simultaneously it is possible to compute

$$a_i = x - r_i, \quad i = 1, 2, \ldots, n,$$

$$b_i = \frac{s_i}{a_i} = \frac{r_i}{na_i}, \quad i = 1, 2, \ldots, n.$$

$$c = \sum_{i=1}^{n} b_i,$$

$$d = \frac{1}{c}, \quad y = d + 1.$$

The development of new numerical methods of solving problems on multiprocessor computers with parallel organization of computations and limited computing resources (certain number of processors, limited volume of memory in each processor, limited number of channels for communication among processors and rate of exchange on these channels, finite number of channels for communication with second-step memory and definite speed of exchange on these channels, fixed length of the machine word, and so on) requires great theoretical (construction of the method, testing its stability, convergence, and so on) and experimental (investigating the efficiency of its realization in various models and applied problems) work, which may run into years.

A way to formulate algorithms that organize parallel computations fairly quickly is to work out new algorithms on the basis of already-known methods. It is

apparent that most algorithms that realize sequential computations permit, in principle, multivariant paralleling. The question is how to construct an algorithm of parallel computations that would be highly efficient. Several criteria can be used to compare and evaluate algorithms: coefficient of accelleration K_y and efficiency coefficient $K_э$, which are defined as follows:

$$K_y = \frac{T_1}{T_p}, \quad K_э = \frac{K_y}{p}.$$

In this case p is the number of arithmetic processes used to solve the problems; T_1 is the time to solve a problem on a single-processor computer with a high-speed arithmetic processor and the total memory of p processors given the availability of the necessary number of peripheral units with exchange speeds as in a multiprocessor computer; T_p is the time required to solve a problem of the same volume on a p-processor computer.

As an example let us consider an algorithm for parallel computation to solve the Cauchy problem for a system of n standard differential equations of the first order

$$\frac{dY}{dx} = F(x, Y), \quad Y(x_0) = Y_0,$$

where $x \in [x_0, x_k]$, Y, Y_0, F are vectors of dimensionality n [3]. As is customary, for a numerical solution we introduce the different network

$$\omega_h = \left\{x_i = ih, \ i = 0, \quad 1, \ldots, N, \ h = \frac{x_k - x_0}{N}\right\}.$$

To solve the problem we will use implicit block methods [4, 5]. Let us review a variation of this method that provides the fourth order of precision. The algorithm of this method for parallel computations can be realized by the formulas

$$Y_{i+1}^{(0)} = \frac{1}{3}(Y_{i-2} + Y_{i-1} + Y_i) + \frac{h}{6}(3F_{i-2} - 4F_{i-1} + 13F_i),$$

$$Y_{i+2}^{(0)} = \frac{1}{3}(Y_{i-2} + Y_{i-1} + Y_i) + \frac{h}{12}(29F_{i-2} - 72F_{i-1} + 79F_i),$$

$$Y_{i+1}^{(s+1)} = Y_i + \frac{h}{12}(5F_i + 8F_{i+1}^{(s)} - F_{i+2}^{(s)}),$$

$$Y_{i+2}^{(s+1)} = Y_i + \frac{h}{3}(F_i + 4F_{i+1}^{(s)} + F_{i+2}^{(s)}).$$

In this i is the number of the point; s = 0, 1, ... is the number of the iteration; and, h is the step of integration.

If there are two processes, the solution is found at point i + 1 in one process and at point i + 2 in the second. Let us note that in this case paralleling is expedient if computation of one right hand part on the average takes less than five operations.

If n is sufficiently large and it is possible to use 2m processors in the multiprocessing machine to solve the Cauchy problem, n/2m equations are placed in each processor.

Table 1 below gives $K_y = \frac{T_1}{T_{2m}}$ and $K_э = \frac{K_y}{2m}$ depending on the number of processors and number of arithmetic operations required to compute the right hand part. This example illustrates the possibility of paralleling computations at the algorithm level.

2 m	Number of Arithmetic Operations									
	5	10	20	50	100	500	1000	2000	5000	10000
2	1.04	1.2	1.4	1.65	1.8	1.95	1.97	1.98	1.994	1.997
	0.52	0.6	0.7	0.82	0.9	0.97	0.98	0.99	0.997	0.998
4	1.57	1.89	2.33	2.97	3.37	3.87	3.92	3.96	3.98	3.99
	0.39	0.47	0.58	0.74	0.84	0.97	0.48	0.94	0.995	0.997
12	2.4	3.08	4.2	6.3	8.1	10.9	11.4	11.7	11.87	11.94
	0.2	0.25	0.35	0.51	0.67	0.9	0.95	0.97	0.998	0.995
24	2.8	3.7	5.3	8.1	12.4	20.1	21.8	22.8	23.5	23.86
	0.11	0.15	0.22	0.33	0.51	0.83	0.9	0.95	0.98	0.99
32	2.9	3.8	5.6	9.7	14.4	25.4	28.3	30	31.1	31.6
	0.1	0.12	0.18	0.3	0.45	0.8	0.88	0.94	0.97	0.987

Table 1.

Paralleling computations at the program level is organized to some degree when working with vectors, even on machines with mainline organization of computations, for example the CDC STAR-100, the CDC-7600 CRAY-1, and some others.

Paralleling at the level of data exchange among processors and between processors and second-step memory is determined by the fact that arithmetic operations are executed more rapidly than exchange operations. Therefore, based on the volume of each processor's main memory, processor speed, and speed of exchange, it is necessary to balance the number of communications channels so that processors do not stand idle because of exchanges while solving problems with large volumes of data to process.

Significant volumes of input and output data require parallel data input/output with monitoring of the reliability of the data fed.

5. In the last 10 years electronic computers with partial and complete organization of parallel computations have been built and put into use. One of the principles by which computers are classified is based on the classification of programs and data processed by the machine. From this standpoint electronic computers can be provisionally subdivided into machines with pipeline organization of computations and multiprocessor computers [6]. A distinction is made among three classes of computers: MISD (pipeline-multiple construction, single data), SIMD (single instruction, multiple data), and MIMD (multiple instructions, multiple data).

It is possible to realize parallel computation of vectors by program means in some computers with pipeline organization. Among these computers is the STAR-100 from the CDC firm. Its speed in performance of vector operations is 100 million ops/sec (6-20 million ops/sec with a floating point). The basic programming language is FORTRAN (the expanded version for work with vectors). The CRAY-1 machine from the Cray Research firm belongs to this class. This machine is designed to solve hydrodynamic problems and forecast the weather. It has eight registers that contain 64 words of 64 bits apiece, and is characterized as a so-called "chain," which makes it possible to use functional units in a given sequence of operations. The speed of the machine is 80 million ops/sec. Another machine that can be included here is the Texas Instruments TI ASC, which is a specialized multiprocessor machine (up to four processors with a speed of 25 million ops/sec). The TI ASC operates with matrix columns and lines and can produce three included cycles with unitary vectors and have four vector channels. The basic programming language is FORTRAN. We should note that computers of class MISD solve fairly complex scientific-technical problems, including several of those mentioned above.

The Burroughs PEPE belongs to the SIMD class of electronic computers.[1] It is a specialized computer consisting of 288 associative processors, each of which has its own memory while data is transmitted through the control processor. The speed of the one central processor is 1-5 million ops/sec. The total productivity is 300-400 million ops/sec. The PEPE can solve high-order systems of linear algebraic equations. Its basic programming language is FORTRAN.

The Burroughs ILLIAC-IV also fits with this class of machines.[2] This computer has 64 processors that work simultaneously. The maximum speed of this machine is 200 million ops/sec. The capacity of the laser memory archive is 10^{12} bits.

---

[1] Bell Telephone Laboratories, System Development Corporation, Honeywell and Burroughs participated in work on the PEPE (Parallel Element Processing Ensemble) program.

[2] The University of Illinois participated in development of this machine.

By 1979 the Burroughs firm planned to complete development of the BSP multiprocessor computer, in the SIMD class, for making scientific-technical calculations. This machine is supposed to have 16 processor elements and, in order to avoid conflicts, 17 memory cubes. The DAP computer of the ICL company also belongs to the SIMD class of computers. The DAP is a matrix of 32 x 32 slow elementary processors, each with a memory capacity of 1,000 bits.

Among Soviet electronic computers, the PS-2000 computing complex belongs to the SIMD class. The PSI-2000 complex includes an SMI-2 small computer and a PS-2000 multiprocessor. The PS-2000 multiprocessor consists of 8-64 processor elements, each of which has its own main memory (from 4,000 to 16,000 24-bit words). The general control unit exercises control over all the processor elements.

The ASC TI can be conditionally classified with the MIMD computers because its four channels can be used for independent data feeding and processing. It is thought that the Siemens SMS laboratory model, which has 128 processors and is now under development, will belong to this class. Each processor has its own memory and can work independently of the others. The processors are joined into one ensemble by the control processor.

Table 2 below shows certain characteristics of these electronic computers. A more complete review of multiprocessor machines and descriptions of them can be found in [8-9].

Despite the fact that multiprocessor computing machines have existed for some 10 years, they have a small share of production. For example, in the United States in 1976 248 models of electronic computing machines were produced, but no more than 10 of them were multiprocessor machines with parallel organization of computations, and half of them were for special applications.[3] They make up an even smaller proportion of world capitalist production of computers. It appears that the most characteristic trend in the production of multiprocessor computers at the present time is to build machines with a large number of processors and limited memory volume for each processor. This structure and architecture complicates problem programming and creates additional difficulties in working out the methods and algorithms for solving applied problems which require a large volume of varied computations.

There is another possible approach to building multiprocessor computers with parallel computations based on the macroconveyor principle developed at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences. This principle of organization makes it possible to solve most of the above-listed scientific-technical problems in a reasonable time.

Theoretical analysis of the different structures and architectures of multiprocessor machines will parallel organization of computations depending on the classes of problems, methods of solution, and volumes of data processed, in addition to work to formulate parallel computation algorithms for

---

[3] We considered here multiprocessor computers with floating points that are known from the literature.

Table 2.

Number	Name of Machine	Date of First Delivery	Cost of Machine (U. S. $)	Addition (Multiplication) Time, μsec	Main Memory Capacity (Thousands of Words)	Average Sorting or Cycle Time, μsec	Word Length, Binary Bit	Additional Information
				(United States)				
I	STAR-100	1974	15 million	0.02	500-1,000	1.28+	64	FORTRAN, APL
II	TI ASC	1973	10 million	0.16	131-1,048 bits integrated microcircuits, 131-1,048 bits MOS structure	0.16+ 1+	32	FORTRAN, 4-8 processors
III	CRAY-1	Oct., 1976	4.5-8.9 million	0.0125	8,000	0.048+	64	80 million ops/sec, 200 million ops/sec
IV	BSP (Burroughs SciPr)	1979	3.9-6.2 million	0.5	8,000	-	48	16 processors, each of which exchanges with its four neighbors, FORTRAN
V	ILLIAC-IV (Simplified version)	1974	40 million	0.3 0.6	two bits, integrated circuit (for one processor)	0.188 0.25+	32, 64	64 processors, speed of 200 million ops/sec, each processor exchanges with four neighbors
VI	PEPE	1977	11.2	-	0.5 for one MOS processor	0.4	32	parallel, FORTRAN, 288 associative processors, 300-400 million ops/sec

[Table 2 continued]

[Table 2 continued]

Number	Name of Machine	Date of First Delivery	Cost of Machine (U. S. $)	Addition (Multiplication) Time, µsec	Main Memory Capacity (Thousands of Words)	Average Sorting or Cycle Time, µsec	Word Length, Binary Bit	Additional Information
				(United States)				
VIII	AP-130	1978	-	-	-	-	32	1,024 processors
				(Great Britain)				
VIII	DAP (Distributed Array Processor)	1976	-	-	1,000 bits for each elementary processor	-	-	experimental model, matrixes of 32 x 32/ 64 x 64 elementary processors, each of which exchanges with four neighbors
IX	CLIP-4	1977	-	-	-	-	-	96 x 96 processors
				(Japan)				
X	HARPS	1976	-	-	-	0.25 for one processor	16	eight processors, full commutator for four processors, and common pipeline
XI	PP	1976	-	-	-	four for one processor	8	16 processors, common pipeline, logarithmic commutator

[Table 2 continued]

[Table 2 continued]

Number	Name of Machine	Date First Delivery	Cost of Machine (U. S. $)	Addition (Multiplication) Time, μsec	Main Memory Capacity (Thousands of Words)	Average Sorting or Cycle Time, μsec	Word Length, Binary Bit	Additional Information
				(France)				
XII	PROPAL-2	1978	-	-	-	16 for one processor	-	128-2,048 processors, each processor connected with two neighbors
				(USSR)				
XIII	PS-200 Computing Complex	1980	-	0.3	4-16	0.64	24	-

hypothetical structures and architectures, permit us to draw some conclusions:

— Paralleling algorithms are defined by solution methods and depend greatly on the structure and architecture of the multiprocessor computer (volumes of main memory of each processor, number of communications channels among processors and between processors and second-step memory, the availability of damper memory, the volume of second-step memory, the ratio of the speed of an individual processor and the volume of its main memory, the length of a machine word, and so on);

— The very same structure and architecture of a multiprocessor machine may give rise to many different versions of parallel algorithms;

— The ease of paralleling an algorithm with sequential computations does not always mean that it is wise to use it as the basis for formulating parallel computation algorithms on a multiprocessor computer, especially for problems with large volumes of computation;

— The critical resources for solving problems with large volumes of data processing are the number of communications channels and speed of exchange in them, the volume of the main memory of each processor (at a given speed) and of damper memory, and the number of input and output units;

— The problem is not so much paralleling arithmetic and logical operations as it is organizing the calculation of problems that require processing large volumes of data, that is, organizing parallel operations at all levels;

— At the present time there is a critical need to work out new theoretically substantiated numerical methods of solving different classes of mathematical problems, techniques whose essential features take account of the parallelism of computations at all levels;

— For each of the classes of mathematical problems considered there is an optimal (with a maximum acceleration coefficient and a maximum efficiency coefficient) configuration of the structure and architecture of the multiprocessor computer depending on the solution algorithm and volume of data processed. Increasing the number of processors beyond the optimal number leads to a certain acceleration of calculation time, but the efficiency coefficient drops.

The necessary configuration of the multiprocessor machine should be shaped, the machine solution algorithm constructed in an automatic mode, the problem solved, and the reliability of the solution obtained evaluated according to the problem, the solution method, and the required volume of data processing, which can be estimated a priori.

This approach to solving applied problems on electronic computers of rearrangeable structure and architecture makes it possible not only to evaluate the quality of solutions, but also to raise the efficiency of the machinery being used.

FOOTNOTES

1. V. D. Ilichev, "Matrix Methods of Synthesizing the Dynamic and Elastic Characteristics of Linear and Conservative Design Elements," UCHEN. ZAP. TSAGI, 1975, 4, No 2, pp 93-108.

2. J. Miklosko, "Synteza a analyza efektivnych numerickych algorimov" [Synthesis and Analysis of Efficient Numerical Algorithms], Bratislava, VEDA, 1979, 220 pages (in Czech).

3. M. A. Franklin, "Parallel Solution of Ordinary Differential Equations," IEEE TRANS. COMPUT., 1978, 27, No 5, pp 413-420.

4. B. A. Rosser, "A Runge-Kutta for All Seasons," SIAM REVIEW, 1967, 9, No 3, pp 417-452.

5. L. F. Shampine, and H. A. Watts, "Block Implicit One-Step Methods," MATH. COMPUT., 1969, No 23, pp 731-740.

6. T. L. Jorden, "A Performance Evaluation of Linear Algebra Software in Parallel Architectures," in "Performance Evaluation of Numerical Software," edited by L. D. Fosdick, Amsterdam, 1979, pp 56-76.

7. F. G. Enslou (editor), "Mul'tiportsessornyye Sistemy i Parallel'nyye Vychisleniya" [Multiprocessor Systems and Parallel Computations], Moscow, "Mir", 1976, 383 pages.

8. V. K. Zeyzenberg (editor), "Vychislitel'naya Tekhnika za Rubezhom v 1976 g." [Computer Technology Abroad in 1976), Moscow, "ITM i VT AN SSSR", 1977, 210 pages.

9. V. K. Zaydenberg (editor), "Osnovnyye Kharakteristiki Tsifrovykh Vychislitel'nykh Mashin" [Basic Characteristics of Digital Computers], Moscow, "ITM i VT AN SSSR", 1977, 63 pages.

11,176
CSO: 1863/16

UDC 62-52.001"313"

PROSPECTS AND PROBLEMS OF DEVELOPING AUTOMATED PROCESS CONTROL SYSTEMS BASED ON MICROPROCESSOR EQUIPMENT

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 8, Aug 81 pp 1-2

[Article by B. B. Timofeyev, academician of the UkSSR Academy of Sciences, in the section "Automated Control Systems"]

[Excerpts] Using microprocessors in the traditional areas of application of computers in automated process control systems is tied intimately to the process of decentralization of control that began long before the appearance of microprocessors. It is characteristic not only of automated process control systems and has two interrelated aspects: territorial distribution and hierarchical control.

The necessity of territorial distribution of control hardware in automated process control systems is dictated to a considerable extent by the deficit in cable production. Remoting of information processing equipment directly to receivers-sources and transmission of compressed, preliminarily processed information are required. Implementation of hierarchical systems under the conditions of continuously growing complexity of control problems allows raising reliability and improving control dynamics, simplifying software, and affording the possibility of putting control systems into operation piecemeal.

Unfortunately, our microprocessor hardware developers are repeating the traditional architecture of computers too literally in their developments when the need for developing multimicroprocessor systems is so obvious. The fact is that transducers make up a large percentage of the hardware of modern automated process control systems and their outputs signals require processing by complex algorithms. These include various frequency, phase (including synchros and inductosyns) and other transducers. The question of connecting up these transducers is tied closely to the problem of developing an efficient architecture of modular microprocessor components. The connection problem for these transducers can be solved by incorporating special-purpose passive functional elements in the composition of microprocessor hardware, i.e. by expanding the nomenclature, which is not always justified economically because batch production declines.

More promising are isolation in automated process control systems and standardization of some simple autonomous regulators implemented by using active programmable modules of modular complexes. These modules must be built on the basis of special-purpose single-board microcomputers that include facilities for input/output of analog and discrete signals.

Progress in this direction depends entirely on progress in LSI and VLSI engineering and technology. This, in turn, requires the joint efforts of the instrument-building and electronics industry to develop a unified modular system of microprocessor control hardware, without in any way reducing the efforts on finishing the development and adjustment of production of existing microprocessor complexes. This unified system is to contain a set of specialized single-board microcomputers, means of normalization and galvanic separation and a ramified set of designs and panel instruments. And this unified system is to allow, by the design route without substantial outlays for engineering labor, developing flexible and highly reliable multimicroprocessor control systems and instrumental process complexes for automating the design of such systems, including highly efficient facilities for programming and debugging.

8545
CSO: 1863/256

UDC 681.2:621.38:776:"313"

PROSPECTS FOR DEVELOPING TECHNOLOGY OF PRINTED CIRCUIT BOARD PRODUCTION IN BUILDING INSTRUMENTS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 8, Aug 81 pp 29-30

[Article by engineers V. R. Truntsevskiy and I. P Belikov, in the section "Technology of Building Instruments": "Printed Circuit Board Production" (beginning of a topical series)]

[Text] Existing technology for manufacturing printed circuit boards [PCBs] and any of its varieties on methods of manufacture are based on the so-called subtractive technology, i.e. the technology of obtaining copper conductors by removing (etching) from blank areas the copper layer covering the substrate. This technology no longer meets the high requirements associated with miniaturization and first of all with broad integration in the field of electronic elements since it does not permit manufacturing minute and precise tracks for conductors [1].

In addition, the subtractive technology of manufacturing PCBs requires making more efficient use of metal because of the growing demand for it. Therefore, process engineers are faced with the problem of finding new engineering solutions that must provide for saving materials and meeting the increased requirements of designers of microelectronic apparatus.

One such engineering solution is the multilayer printed circuit assembly which, being a subsequent qualitative evolution of conventional PCBs and preserving all their properties, have their own merits that promote an ever broader application of them in solving the problems of complex miniaturization of microelectronic apparatus. According to foreign data, as early as in 1973, almost half of all PCBs used in computers were multilayer PCBs (MPP).

Multilayer PCBs now have limited application in the Minpribor [Ministry of Instrument Making, Automation Equipment and Control Systems]. The increased technological requirements that occur in the production of multilayer PCBs, the lack of mobility of the technological process, the extremely long production cycle in manufacture, the labor-intensive monitoring of the internal layers and limited repairability are largely restraining the application of multilayer PCBs in the products of instrument manufacture. However, computer hardware developers are now tasking process engineers with assimilation in the Minpribor of the technology of manufacturing multilayer PCBs in connection with the need of applying them in new products.

In the 11th Five-Year Plan in the plants of the Minpribor, the main development will be on the combined positive method with advance plating of holes and a protective technological tin-lead alloy coating. Dry alkaline-water-developed film photoresists of the types TFPK, SPFV and SPFVShch, as well as SPF-2 film photoresists processed in organic solutions, will find extensive application in the process of photochemical printing to obtain the printed circuit pattern. This method will permit manufacturing printed circuit boards with a density of placement of 150-300 holes per dm^2.

Since in the early years of the 11-th Five-Year Plan the demand by Minpribor plants for dry film photoresists cannot be fully met, it is planned to use the FPP type liquid photoresist which will be applied to blank boards with advance plated holes on U861M type rolling units. In the process, the necessary thickness of the photoresist layer is achieved.

Efforts are now underway to improve this technology. Thus, it is planned to eliminate the operation of chemical metallization of holes from the process which will not only eliminate the use of a critical metal (palladium dichloride), but also reduce the manufacturing cycle and raise quality.

The semiadditive method of manufacturing PCBs has been developed in the sector to produce PCBs with a density of placement of 400 holes per dm^2 and higher (with a 0.1-0.2-mm width of printed conductors).

This technology differs from the existing subtractive in the application of the non-foil dielectric STEK-1.5 TU.1MO.50.9.091-78 and in the considerable reduction in the volume and time of etching, which makes it possible to obtain substantial savings in copper, reduce the width of printed conductors (having eliminated etchings of them) and reduce costs for purification structures. On the whole, there will be a 5 to 20 percent gain in value compared to the subtractive technology [1].

This technology will be developed along the path of mastering the so-called photoform method and making use of dielectrics with ultrathin foil (to 5 micrometers). It is assumed that when the photoform method is used to manufacture boards, production costs will be 25 to 30 percent lower than when photoresist is used. This method allows obtaining a higher density of circuitry with the same board technological characteristics.

An advantage of semiadditive technology is the capability of reusing laminates when defects occur in manufacture by stripping the circuit and reapplying the conductors.

The firm Photcircuits estimates that 0.5-0.8 million m^2 of boards per year are now produced in the United States by the method of semiadditive technology [2]; major American manufacturers of ultrathin foil estimate that boards in which ultrathin foil is applied now account for 5 to 10 percent of the total PCBs produced in the United States [2].

Efforts are also underway in our country to develop a dielectric with ultrathin foil. It is planned to start series production of it in the 11th Five-Year Plan [3]. Assimilation by Minpribor plants of the combined positive method and semiadditive technology of PCB manufacture will allow shifting subsequently to the assimilation of new types of these technological processes without any restructuring of production.

The ever increasing complexity and continuing improvement of instruments and computer hardware is leading to the necessity of making various adjustments to wiring layouts even under the conditions of series production. But making changes to a printed circuit is a lengthy process. According to the data of foreign specialists, the share of expenses for developing a board designed for wiring of 80 integrated circuits (IS) is about $7,000 [4]. This sum of fixed expenses is due to the participation in the implementation of a considerable number of skilled associates and the lengthy duration of the manufacturing cycle. Therefore, for small-scale series production of high-density wired boards and accelerated development of prototype instruments and computer hardware, application of fundamentally new manufacturing methods is called for: thin-conductor wiring (of the multiwire type) or K-6 technology which have the same advantages of printed circuitry but are free of the shortcomings of the subtractive technology, are technologically less labor-intensive, and afford a reduction in the requirements for precision of manufacture and materials used; but the main thing is that the production cycle is shorter and they are more suitable for automation of design and manufacture.

The thin-conductor method of wiring in practice can be based only on dry processes. The method precludes the use of photomasks and the process of photochemical printing and etching.

Multiwire circuitry allows successfully replacing multilayer PCB's when circuitry is highly complex.

Foreign data indicate that the design cycle is 10 weeks for multilayer PCBs, 8 weeks for double-sided plated boards, but just 1 week for multiwire; manufacture of the boards takes 4, 2 and 1 week(s) respectively [4]. Approved designs of boards manufactured by the method of conducting wiring, when series production is necessary, can easily be switched to the traditional technology of printed wiring and in a shorter time. According to data of foreign specialists, fixed costs in manufacturing boards with 80 integrated circuits by one of the methods of conducting wiring (solder-wrap) are $700-$1070 [4]. Thus, it is advantageous to employ methods of conducting wiring with small-scale series production (especially experimental) where the capability of making changes without additional outlays is exceptionally important.

The technology of thin-wire circuitry will be developed with the solution of the following problems:

improvement of the method for obtaining contact connections;

determination of the optimal technological conditions of layout of conductors, limitation and features for achieving maximal productivity of automated manufacture of conducting wire boards;

development of algorithms for the process of automated layout of conductors for numeric controlled machine tools equipped with special layout devices; and

development of special machine tools with ChPU [numerically programmed control] for (pistonirovaniye) and layout of wires.

All the efforts planned to be performed in the 11th Five-Year Plan have been incorporated in a comprehensive program for raising the level of the technology for producing PCB's; its implementation by the end of the five-year period will allow obtaining an economic effect of 25.2 million rubles and conventionally releasing over 2,900 personnel.

In subsequent articles the authors will go into more detail on specific questions associated with the technological processes of manufacturing PCB's and the special technological equipment provided for by the comprehensive program.

BIBLIOGRAPHY

1. Kaganov, Ye. A.; Kochetov, Ye. P.; Kravchenko, A. S. and Maksakov, E. P., "Additive Methods of Manufacturing Printed Circuit Boards," ZARUBEZHNAYA RADIOELEKTRONIKA, No 6, 1974.

2. RADIOELEKTRONIKA ZA RUBEZHOM, No 18, 1978.

3. INFORMATSIONNYY LISTOK No 235-79, LMTs.

4. RADIOELEKTRONIKA ZA RUBEZHOM, No 9, 1980.

8545
CSO: 1863/256

UDC 621.38:776

EXPERIENCE OF ASSIMILATION OF BASE TECHNOLOGY FOR PRINTED CIRCUIT BOARD MANUFACTURE AT LENINGRAD ELECTROMECHANICAL PLANT PRODUCTION ASSOCIATION

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 8, Aug 81 pp 30-31

[Article by engineer G. M. Tsetkov]

[Text] The problem of assimilating production of automation equipment and fourth generation computers with the application of LSI, VLSI and microprocessors is inextricably entwined with the broad introduction of the positive combined method of manufacture that GOST 23751-79 recommended as the base technological process for production of printed circuit boards [PCB's].

Discussed in this article are the features of the new technology for PCB manufacture by the combined method with advance drilling of holes and use of film photoresist and positive screen printing, as well as the experience of assimilating and implementing this technology in series production on domestic equipment at the enterprise that was one of the first in the sector to practically resolve this problem, the PO LEMZ [Leningrad Electromechanical Plant Production Association].

The progressiveness of the new technology is generally recognized and stems from the capability of manufacturing PCB's with increased density of wiring, improved quality and reliability, less manual labor, and broader application of equipment for automation and mechanization of production. But, as is well known, this process is being introduced into series production slowly.

The base method for PCB manufacture was introduced in the Leningrad Electromechanical Plant Production Association in 1977. The necessary prerequisites for its introduction were created thanks to the large amount of laboratory work and engineering preparation of this process. Thus, PCB's were standardized and their type sizes reduced to two main ones: 1.5 x 110 x 140 mm (type A boards) and 1.5 x 140 x 235 mm (type B boards); photomasks of the manual layout were processed and manufactured by the more precise machine method on the "Minsk-2004" and "Minsk-2005" coordinatographs.

The V-479 type domestic machine tools with program control were modified and are now used to drill the PCB's. The boards are drilled in batches of three to four units, and the last board is not drilled, but placed first in the next batch to reduce burrs at the drill entry and exit. Drill bits are hard alloy twist with a durability of no less than 2,500 holes. Holes are not counterbored. If the size of burrs exceeds 0.04 mm, then smoothing is necessary, which is done on a unit for

hydroabrasive trimming of the holes. Chemical preparation of the surface in a complex with mechanical smoothing is performed on the domestic U759M type jet-module line.

In chemical metallization, a combined solution of activation containing 0.2-0.4 g/l of $PdCl_2$ is used. The stability of operation of the combined solution is 5-6 months. The solution is analyzed and adjusted twice a week.

After activation, the boards stand in a solution of accelerator (250 g/l HCl).

The solution of chemical copper plating is prepared on the basis of Seignette salt with a low-concentrated content of the main components with a stabilizing additive of thiosulfate of sodium. Twice a day the solution is analyzed and adjusted for the basic components, and filtered once a shift. The stability of operation of the solution of chemical metallization is 1.5 months.

Sulfuric acid electrolyte of copper plating is used for preparatory electroplating by 5-7 micrometers. Bonding strength of the chemical copper to the foil is 400-500 gs/cm (as defined by OST 4G0.054.223).

Prior to application of the film photoresist, preparation of the surface of the copper layer is required; this consists in mechanical conditioning of it by wire brushes made of 0.15-mm diameter stainless steel, cleaning in a two-percent alkaline solution, washing and drying in a special unit.

A dry film water-developed photoresist, brand SPFV (OAYu.504.022TU), is applied with a KP6346 type laminator and the U862 type unit with 110° C temperature of the heating elements. The blank boards are fed without preheating. The effective time of exposure to the LUF-80 lamps is 3-4 minutes, and to the DRST-1000 lamps is 1.5 minutes. The photolayer is developed in the jet-modular unit by a one-percent soda ash solution.

The coating of copper and tin-lead is applied in standard electrolytes in automatic electroplating machines. A Kh606 type ampere-hour meter is installed in one of the tanks.

The photoresist is removed in a ten-percent alkaline solution at normal temperature within 3-5 minutes. The problem of salvaging the spent solution and scraps of film completely has not yet been solved. Etching is done in an ammonium peroxysulfate solution.

Palladium plating of the thin strip leads is done directly on the copper printed conductor by the group method in a special unit in standard aminochloride electrolyte. For the palladium coat, the surface is prepared by chemical and mechanical methods. The copper conductors are polished in a special G6-228 type unit.

A method of positive screen printing has been assimilated with 15 board positions. Used for screen printing is a single-component, alkali-washable, STZ.12-51 brand,* yellow ink that after drying at 90-100°C (within 20-30 minutes) acquires resistance to acid electrolytes and cures the action of etching solutions based on cupric chloride, ammonium persulfate and ferric chloride. The ink has fine print

* (TU 29-02-740--77)

properties and is applied with manual masking machines and with PTP-3 type semi-automatic machines using polyurethane squeegees. Blue ST3.13 brand ink (TU 29-02-489--74) was also tried for positive screen printing. The ink is resistant to acid and alkaline electrolytes and dries in 8-10 minutes with a hot-air (60°C) blower or infrared dryer within 1.5-2 minutes at 100-110°C in the operating zone. However, this ink has an important shortcoming: Only organic solutions can wash it cff.

To preclude conductor peeling in the PCB production process, before applying ink, the surface has to be treated with quartz to improve ink adhesion to it or electro-polishing of the boards has to be introduced during chemical copper plating.

Ink layer thickness is 30-40 micrometers. In determining the effect of the basic factors of the printing process on the graphic characteristics of the image, the optimal conditions of pressure, rate of printing and slope of the squeegee must be established.

Screen masks are made by the direct method using the liquid photopolymerized composition "Fotoset-Zh" (TU 6-15-01-138--77).

The technological process of making mask print forms is as follows: a form frame with the screen tight on it is placed on glass coated with lavsan [Soviet equivalent of Dacron], "Fotoset-Zh" is poured on center of screen, it is covered with the photomask (previously coated with a five-percent solution of paraffin in white spirit), smoothed out in thickness using a squeegee, and the excess solution is removed with a pad. The layer on the screen must be even, without bubbles or wrinkles. Then the form is exposed by LUF-80 lamps for 15-20 minutes.

The time for making forms is reduced considerably with this technology. This method allows regulating the [illegible]ness of the copy layer. The photopolymerized composition ensures obtaining forms with clear edge of the printing element.

Repeated use of screen masks is anticipated because they are made on screens of stainless steel No. 004. The precision of screen printing ensures obtaining a pattern with width of conductors and distance between them of 0.3-0.4 mm.

On the whole, implementation of the base technological process for PCB manufacture has made it possible to:

improve the quality of drilling of holes and chemical metallization of the boards since the operations are performed on blanks not coated with protective lacquer; in the process, controlling the quality of performance of these operations is eased considerably;

extend the working life of the solutions for chemical metallization;

accelerate the manufacturing cycle by removing a number of operations for photo-printing processes (coloring, chemical hardening, thermal hardening) and application and removal of the protective lacquer;

reduce manual labor thanks to mechanization and automation of the operations of application, development and removal of film photoresists and reducing to the minimum the necessity of performing the retouching operation;

almost completely eliminate technological losses in photoprinting operations, which accounted for about 30 percent of the total losses when a light-sensitive emulsion based on polyvinyl alcohol was used; moreover, the capability of efficiently repairing rejects has emerged with full recovery of the labor spent on hole drilling;

increase production efficiency and reduce the labor-intensiveness of PCB manufacture by 25-30 percent; and

increase the density of wiring of the printed elements to 0.2-0.3 mm with improvement of the quality of the finished boards, which is confirmed by the objective test made on the special automatic machine for checking the circuitry of printed boards developed at the Leningrad Electromechanical Plant Production Association.

8545
CSO: 1863/256

SOFTWARE

LIST OF SOVIET ARTICLES DEALING WITH SOFTWARE

Moscow ALGORITMY I PROGRAMMY in Russian No 6, Jun 81 pp 1-126

[Following is a listing of Soviet entries from ALGORITMY I PROGRAMMY ("Algorithms and Programs"), a bibliographic publication of the USSR State Committee for Science and Technology. This listing is from No 6, 1981]

[Excerpts]

2297. "Teoreticheskiye osnovy kompilyatsii" [Theoretical Fundamentals of Compilation]: collection of scientific works, Novosibirsk State University, Novosibirsk, 1980, 171 pages, bibliography at end of articles.

Study of the theoretical fundamentals and applied aspects of the formal description and effective implementation of programming languages.

Key words: method, PASCAL, MEDIFOR-2 knowledge display language, RELYAP relational programming language, LISP, ALGOL, M-4030, YeS computer, R6060 minicomputer, macrogenerators, KAMAK [CAMAC], physical oceanographer's package of applied programs, single-variable polynomials, program correctness, language processors, compilation.

2302. Sychev, A.I. "One Method of Automatically Controlling an Archive of Meteorological Information," TRUDY ZAPADNO-SIBIRSKOGO REGIONAL'NOGO NII, VYP. 49, VOPROSY REGIONAL'NOY KLIMATOLOGII I MEKHANIZATSII OBRABOTKI METEOROLOGICHESKOY INFORMATSII [Works of the West Siberian Regional Scientific Research Institute, No 49, Questions Relating to Regional Climatology and Mechanization of the Processing of Meteorological Information], pp 115-117, references 7.

Creation of a filter for detecting gross errors based on taking into account the change in meteorological elements during adjacent observation periods, detecting certain maximum deviations and the dates corresponding to them, and implementing this in ALGOL.

2313. Zlobin, V.A., Sverdlov, L.Z. and Shadrin, A.D. "Use of a Computer in Performance of Work on the International Certification System," ELEKTRONNAYA TEKHNIKA, SER. 8, UPRAVLENIYE KACHESTVOM, METROLOGIYA, STANDARTIZATSIYA [Electronic Engineering, Series 8, Quality Control, Metrology, Standardization], No 6 (84), pp 64-68, references 6.

Three groups of statistical problems are described which originate in preparing for certification ferrite products, in conformity with the requirements of the International Certification System (MSS). The software of problems of the first and second groups is implemented in ALGOL. The third group of problems concerns the simulation of plans for statistical acceptance testing and the establishment of an optimum plan for introduction into specifications for the MSS.

2379. Babayev, T.A. "Algorithm for Optimum Distribution of Territorial Centers for Complete Centralized Servicing," VOPROSY RADIOELEKTRONIKI, SER. EVT [Questions Relating to Radio Electronics, Computer Technology Series], 1980, No 6, pp 77-84, references 1.

An algorithm in FORTRAN can be used in planning and in expanding the existing complete servicing system.

2403. Zor'yan, L.B. and Mishchenko, Ye.P. "Determination of the Rate of Scanning of a Specific Sector by a Tracking Radar," VOPROSY RADIOELEKTRONIKI, SER. OT, 1980, No 4, pp 73-78, references 1.

An algorithm in FORTRAN for determining the rate of motion of the electrical axis of an antenna in terms of azimuth as a function of the dimensions of the scanning area, the time an aircraft appears in this area, the width of the directional pattern, the size of its rake angle and the scanning rate for tracking radars.

2458. Gorskiy, L.K. and Golev, V.P. "Control of Complicated Process Organization Systems Based on Imitative Simulation," VOPROSY RADIOELEKTRONIKI, SER. EVT, 1980, No 6, pp 8-19, references 2.

A description is given of FORTRAN programs subdivided in terms of purpose into four groups: preprocessing of source data and discovery of logical errors; imitative on-line control; processing of results of simulation by methods of mathematical statistics and production of output documents; service programs for carrying out a dialogue with the computer.

2491. Kukharenko, B.G. and Chironov, V.V. "Simulation of a Complete Centralized Servicing System," VOPROSY RADIOELEKTRONIKI, SER. EVT, 1980, No 6, pp 61-66, references 3.

A program in GPSS is presented for the imitative simulation of a regional service center which receives information on computer failures.

2494. Volkov, N.V. and Muzykin, S.N. "Study of Nonlinear Systems in the Normal Functioning Mode" in "Algoritmicheskiye metody i programmirovaniye v radioelektronike" [Algorithmic Methods and Programming in Radio Electronics], inter-VUZ collection of scientific works, Ryazan' Radio Engineering Institute, Ryazan', 1980, pp 27-35, references 2.

A method of identifying nonlinear systems by adding white noise to the legitimate signal. A description is given of expansion of the output signal into a series of orthogonal Wiener functionals.

Key words: method, FOKAL, Elektronika-100/I, nonlinear systems, Wiener method, identification, output signals.

2573. Drovozyuk, V.S. "Centralized Introduction of 'OKA' and 'KAMA' Packages of Applied Programs," VOPROSY RADIOELEKTRONIKA, SER. EVT, 1980, No 6, pp 67-71.

2580. Khasanov, I.A. "The 'Controller' Program--a Means of Dialogue Debugging of Programs Written in Assembly Language," VOPROSY RADIOELEKTRONIKI, SER. EVT, 1980, No 6, pp 120-126, references 2.

The program is implemented in the ASSEMBLER language and its capacity is about 20 K bytes. The program's input language is described.

2581. Tsenilov, G.A. "Technological Aspects of Applied Programming for the 'KAMA' System," VOPROSY RADIOELEKTRONIKI, SER. EVT, 1980, No 6, pp 20-34, references 11.

Questions relating to the technology of applied programming based on the operation approach to designing and developing automated real-time information systems based on the KAMA package of applied programs.

2582. "Tsifrovyye ustroystva i mikroprotsessory" [Digital Equipment and Microprocessors], Latvian SSR Academy of Sciences Institute of Electronics and Computer Technology, No 4, Riga, Zinatne, 1980, 172 pages. Bibliography at end of articles, ISSN 0137-0480.

Questions relating to the design of microprocessor systems and methods of interfacing them with real entities in subsystems for controlling electric trains and motor vehicles, individual units of microprocessor equipment and digital automation systems, the development of software by means of cross software, and simulation of operation.

2590. Bober, L.N., Bober, T.N. and Morev, V.N. "Procedure for Training Specialists in YeS [Unified Series] Computers by Using Educational Television," VOPROSY RADIOELEKTRONIKI, SER. EVT, 1980, No 6, pp 95-100, references 5.

2656. Viktorov, L.P. "MIM--Monitor dlya Interaktivnogo Modelirovaniya" [The MIM--Monitor for Interactive Simulation], preprint, Moscow, 1980, 71 pages, on superheading: All-Union Scientific Research Institute of Research Systems, references 5.

The MIM dialogue system is used on a PDP 11/70 computer when working with models of global development and other complicated imitative models. The system's instruction language assigns various scenarios, controls the simulation process and outputs the results in graph and table form. The size of any problem is less than or equal to 64 K bytes.

2664. Dul'ko, Ye.N., Dyad'kin, I.G., Kalugin, V.A. et al. "Complex of Software and Constants for Problems in Nuclear Geophysics on the M-222" in "Matematicheskoye modelirovaniye v yadernoy geofizike" [Mathematical Modeling in Nuclear Geophysics], USSR Academy of Sciences Bashkir Branch, Ufa, 1979, pp 157-173, references 13.

General software programs and programs for simulating the propagation of neutrons and gamma quanta in matter in BM-4/220 autocode; standard programs for the draw game

for the range of the path of particles in a system of media with cylindrical interfaces are written in FORTRAN in view of the branched logic.

2666. Luzgin, V.S. and Sushkov, O.G. "One Method of Calculating the Automated Control System Which Is Optimum in Terms of Efficiency," TEKHNIKA SREDSTV SVYAZI, SER. TEKHNIKA RADIOSVYAZI [Communications Equipment Engineering, Radio Communications Equipment Series], 1979, No 9 (26), pp 103-109, references 2.

An example is given of the use of a method for calculating on a YeS-1033 computer an antenna matching unit which operates at a fixed frequency. Calculation time is about 2 h.

2669. Savel'yev, P.V., Zhuravlev, V.Ye. and Dubonos, S.V. "Program for Statistical Analysis and Estimation of the Reliability of the Results of Parametric Optimization of Integrated Circuits" in "Fizicheskiye osnovy mikroelektroniki" [Physical Fundamentals of Microelectronics], collection of scientific works, Moscow Institute of Electronic Engineering, Moscow, 1979, pp 148-157, references 3.

A description is given of the algorithm and structure of the STANAL program which, based on a statistical analysis, simulates the process of the fabrication of integrated circuits and predicts the yield of suitable ones and estimates the reliability of the results of parametric optimization of components. The STANAL program is written in FORTRAN for the "Dubna" simulation system of BESM-6 computers and the disk operating system of YeS computers.

8831
CSO: 1863/22

UDC 681.3.06.001.57

MODELING PROGRAM DEVELOPMENT AND MAINTENANCE AT COMPUTER CENTERS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 8, Aug 81 pp 3-5

[Article by A. P. Ivanov, doctor of economic science, and N. V. Tychina, engineer, in the section "Automated Control Systems"]

[Excerpt] The widespread use of third-generation computers has led to the emergence of a qualitatively new type of ASU software: application program packages (PPP). The industrial approach to the development of programs means that the process of creating them assumes the nature of manufacture of an industrial product, and dissemination--the nature of industrial services. The scale of PPP development in our country and the universal use of YeS computers in control systems have led to the need of establishing a specialized organization--the NPO [Scientific Production Association] "Tsentrprogrammsistem" (in Kalinin) whose main task is software maintenance.

The term "maintenance" implies a broad range of work and services that the NPO "Tsentrprogrammsistem" performs on economic contract bases for enterprises and organizations that are implementing ASU's. By mid 1979, the central stock of algorithms and programs for automated control systems (TsFAP ASU) at the NPO "Tsentrprogrammsistem" was about 150 application program packages with a total size of over 5.92 million machine instructions. In 1978 alone, about 2,200 PPP's from over 600 organizations and enterprises in the country were turned over for maintenance by the specialists of the TsFAP ASU.

By maintenance of application program packages, the following work and services are generally understood: rendering assistance to users and consultation on choosing design solutions during operation of programs; obtaining working programs with adjustment for user parameters, finding and eliminating errors in the text of programs, creating new versions of the latter, modification of their functional capabilities; adaptation to the computer hardware and software used; changing and reissuing user documentation for operation of systems.

8545
CSO: 1863/256

UDC 658.51.012.011.56:[681.3.06:519.256

CREATION OF SOFTWARE BASED ON USE OF PACKAGES OF APPLIED PROGRAMS

Moscow MEZHOTRASLEVYYE VOPROSY NAUKI I TEKHNIKI, OBZORNAYA INFORMATSIYA: SOZDANIYE PROGRAMMNOGO OBESPECHENIYA NA BAZE ISPOL'ZOVANIYA PAKETOV PRIKLADNYKH PROGRAMM in Russian No 7, 1981 (signed to press 7 May 81) pp 5-9, 18

[Book entitled "Creation of Software Based on Use of Packages of Applied Programs" by Vladimir Sergeyevich Rumyantsev and Nataliya Yakovlevna Lopatkova, GOSINTI, 3160 copies, 41 pages]

[Excerpts] In the last decade a large number of programs, packages and program systems have been developed in this country. Organizing the accumulated store of software is one of the most acute problems in the contemporary stage. Each enterprise, in developing ASU [automated control system] software had to independently determine for itself the only correct variant of use of the ASU software fund available in the country.

For example, in the development of the software of the branch ASU of the USSR Ministry of Construction Industry such packages of applied programs (PPP) are used as: PPP-GVV [generator vvoda-vyvoda--input-output generator]; PPP-LPZ [lineynogo programmirovaniya--linear programming]; PPP-bukhalterskiy uchet [accounting]; PPP-konvertor [converter], and for accomplishment of control functions--programs of original development. At the same time, at the Main Computer Center of the USSR Ministry of Construction Industry a branch fund of algorithms and programs has been created which includes programs developed at enterprises of the branch which are of interest to software developers at related enterprises.

At the "Energopribor" experimental plant (Moscow) the software developers--the State Scientific Research Planning Institute for Introduction of Computer Technology into the National Economy (GNIPI VT, Kazan') and the "Tsentrprogrammsistem" Scientific Production Association (Kalinin)--were oriented toward use of an information system for production control. As a result of examination of the enterprise and the execution of work on the attachment of packages of the information system for production control, at the plant they gave up use of the PPP-UZ [upravleniye zapasami--stores control) and the PPP-PM [planirovaniye moshchnosti--capacity planning] is used partially. In addition, many programs of the usable packages of the system were modernized (for example, in part of the change of the horizon of planning) and supplemented by programs of original development.

For the "Groznefteorgsintez" (Groznyy) software the software of the ASUP [avtomatizirovannaya sistema upravleniya predpriyatiyami--automated enterprise management

system) also was developed on the basis of the ISUP system, but here they gave up use of the package of the SIOD [sistema informatsionnoy obrabotki dannykh--integrated data processing system] as a result of complete change of the package printing of the output forms.

At one of the ship-maintenance plants the PPP-PMOU [planirovaniye moshchnosti i operativnoye upravleniye--capacity planning and operative control] was selected for execution of the function.

The "Lenelektronmash" Scientific Production Association (Leningrad) is developing software on the basis of the "Zapros" system for the undertaking of basic tasks implementing the automated enterprise management system.

Thus, some enterprises develop software using individual standard modules and programs, others, packages of applied programs; still others--both; others use program systems (completely or partially, selecting packages or individual programs from the PPP); still others link scattered packages into a single program system and supplement original developments, etc. As practice shows, in the process of attaching standard software they are modernized, modified and corrected, and this expands the circle of potential users of these means of automation.

The use of standard software in ASU planning assures a reduction of the time, cost and labor-intensiveness of ASU creation for each specific purchaser, a high degree of reliability of the software, a systematic nature and efficiency of introduction of changes in the ASUP software to meet requirements of the purchase, and also creates possibilities of exchange of experience for the purpose of teaching the personnel the work under the conditions of functioning of the ASU, exchange of programs for the solution of problems of control and data processing, the cooperation of user organizations to create general classifiers of information files and methodical materials.

The results of investigating directions of standardization of ASU special software [12,13] permit the conclusion that the subsystem method has a higher degree of integration of elements than the elementary method and therefore the process of creating ASUP software on the basis of them is more effective. In recent time an increase in the use of packages of applied programs is noted, and this is explained by:

--constant increase of the software fund (Table 1), and in particular of the PPP in the TsFAP [tsentral'nyy fond algoritmov i programm--central fund of algorithms and programs]--in the last 3 years alone the number of PPP for YeS computers has become 3.3 times as large;

--low cost of acquiring packages as compared with financial and labor expenditures on developing them. Usually the specific cost of a machine instruction of an individually developed program is 4-5 rubles. The "Tsentrprogrammsistem" Scientific and Production Association sells programs of the fund at a price of 3 kopecks, which together with expenditures on the organization and introduction of programs is reduced to a third or a half of expenditures on individual development [9]. In addition, in the USSR the cost of PPP acquisition is 1 percent of expenditures on its development (in foreign countries, about 10 percent) [4]. The labor-intensiveness of the development of ASUP software with the use of ISUP PPP, for example, is reduced by about 30 percent on the whole;

Table 1

Year	Volume of fund: Number of programs (items)	Volume of fund: Number of machine instructions (1000 items)	Number of organizations using ASU TsFAP
1974	11	408	42
1975	34	2057	228
1976	64	2909	354
1977	104	4300	516
1978	150	5600	600
1979	180	6300	700
1980	200	9000	1000

--a shortage of qualified specialists in the area of programming.

The use of PPP permits ASUP developers to have a set of working programs which service specific needs; to shorten the time taken to obtain working programs from the start of planning and reduce the cost of work on programming (if the PP has already been organized); to exclude the step of posing most problems in the development of a technical plan; to vary the programming and debugging of programs by selecting parameters of generation in the corresponding language of the package and the generation of working programs.

PPP users are attracted by simplicity of operation, flexibility (the possibility of introducing changes without substantial expenditures), reduction of the number of specialists, who form the intermediate link between personnel management and the data processing system.

The saving from introduction of software of the centralized fund of the "Tsentrprogrammsistem" Scientific Production Association in 1979 was more than 20 million rubles, but in that year the association was able to satisfy only half the demands of organizations and enterprises of the country for the delivery of programs. This is a result of limited production capacities of the association and an inadequate quantity of software in the ASU TsFAP.

To further develop the industry of production of ASU software, in the Tenth Five-Year Plan organizations of various ministries and departments of the country, under the leadership of the USSR State Committee for Science and Technology, carried out a complex program of work on the creation of over 100 ASU programs for mass application, and in the Eleventh Five-Year Plan a further expansion of the fund of algorithms and programs is proposed. In connection with that a need arises in ASUP users and developers for improvement of the forms and methods of control and dissemination of the created software.

The successful solution of that problem ought to improve the quality of the software developed, to contribute to the standardization of work on the use of hardware to create conditions for software interchangeability, to assure the interest of software developers in repeated use of their product to supplement the fund of applied programs, and also to further modernize the existing software and to increase

Table 4 Comparative characteristics of hardware and software of packages

Characteristics	ISUP	SPO-Problema	PPP-OUOP	PPP-PMOU
Composition	2 general-purpose PPP's (information) 9 functional-purpose PPP's	1 general-purpose PPP 8 functional-purpose PPP's	1 functional-purpose PPP	1 functional purpose PPP, including 9 complexes consisting of one or several components
Hardware (not counting that needed for work of operating system):				
computer series	YeS	YeS	YeS	YeS
operating system	DOS OS	OS version 4.1	OS version 4.1	DOS OS
main memory volume, Kbytes	256 256	512	128	64 128 (MGT) or 256 (MVT)
number of disk drives	2 2	6	2	3 1 (29 mb) or 3 (16 mb)
number of magnetic tapes	- -	4	2	1 -
Programming language	Assembler	Assembler, PL/1, COBOL	Assembler, COBOL	Assembler
information base	SIOD 1, 2, 3 SIOD OS SIOB 1, 2	SUBD OKA, PPP-SR (system solution)	User base; PPP of UNIBAD family	Base is formed by one of complexes of the package

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000500010023-0

the efficiency of ASU by standardizing work on the planning and technology of software creation.

At present work is being done in that direction by the "Tsentrprogrammsistem" Scientific Production Association (Kalinin), the "Lenelektronmash" Scientific Production Association (Leningrad), the GNIPI VT (Kazan'), the Scientific Research Planning Institute of Automated Control Systems (NIPI ASU, Volgograd) and other organizations.

In practice a variant of software development is selected by the ASU user and developer together. An analysis shows that the quality of the solution of that question depends to a great degree on the level of occupational training and qualifications of the participants in creation of the ASU and, in particular, on the qualifications of the programmers. For example, if at an enterprise specialists who have created a software system for another enterprise are acting as the developers, then, as a rule, the variant of software development is repeated. Therefore improvement of the organization of control of the fund of algorithms and programs and, in particular, the development of a system of estimates of the readiness of all standard software (primarily PPP) will permit determining the rational variant of software creation under the specific ASUP conditions.

The PPP represents a system characterized by a technical and information base and program and organizational softarre. As a result of the fact that each package is oriented toward one class or another of computing systems, the time and cost characteristics of PPP adaptation and operation will depend to a great degree on the technical base existing at the enterprise developing the ASU (the computer configuration, the operating system, the interactive facilities, the computer mode of operation, the stores, etc). Each PPP assumes the presence at the enterprise of definite forms and methods of information organization and its collection, preparation and perforation; in addition, as a rule, the information carriers, the file formats and characteristics, the operational and flow technology of proessing, etc, are fixed. The entire aggregate of software of a given package has a strictly determined architecture reflecting the informational and functional relations of the programs in the PP (programs function, of organization and serving without data, input-output, of data monitoring and transmission, organizing, etc). In the dissemination of packaging, besides transmission of the program texts (print-outs of them on machine media) a large volume of work is done in connection with improvement of documentation for the PPP (besides correspondence of the documentation to the standard it is necessary to develop methodical guides on the attachment and use of the package), maintenance, instruction of users in work with the package and other types of work.

2174
CSO: 1863/33

UDC 51:681.3:007

MULTILEVEL STRUCTURAL DESIGN OF PROGRAMS: FORMALIZATION OF THE METHOD — SPHERE OF APPLICATIONS

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 42-65

[Article by V. M. Glushkov, G. Ye. Tseytlin and Ye. L. Yushchenko]

[Excerpts] Intensive studies on programming technology are underway today.[1] Work [11] gives a thorough analysis of the basic problems that arise in this field and notes the significant contribution to its development made by the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences: the method of formalized technical specifications [18, 19], R-technology [5, 6], and structural programming based on systems of algorithmic algebras (SAA) [10, 22]. The apparatus of SAA proposed in 1965 is adequate to the conception of structural programming which has received universal recognition and broad distribution in recent times [27, 28, 35, 52]. The distinguishing characteristics of the SAA apparatus in comparison with other well-known algorithmic systems (for example [32]) lies in the possibility, owing to its programming orientation, of solving the problem of a uniform formal description of the structures of the algorithms and the programs associated with them in the process of their joint development. It should be noted that this problem is comparable in significance to the problem of joint designing of the hardware and software of contemporary computers [15].

Further development of the mathematical foundations of structural programming was embodied in the method of multilevel structural designing of programs [56, 23], which combines modified SAA [54, 22] with formal models of languages [9, 20].

The development of programs using the multilevel structural program design method consists in a formal description of them in terms of regular SAA charts at each level of design and formalization of the process and level-by-level transfers by constructing an outlet in the corresponding grammatical model. Another important advantage of the multilevel structural program design technique

---

[1] See "Tr. I. Vsevoyuz, Konf. po Tekhnologii Programmirovaniya. Kiev-79" [Works of the 1st All-Union Conference on Programming Technology. Kiev-79]; KIBERNETIKA, 1980, No 2; PROGRAMMIROVANIYE, 1980, No 2; UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, 1980, No 2.

is that it is coordinated with other well-known programming technologies [5, 19, 50, 34]. This makes it possible to use this technique in the highest stages of development of algorithms and programs, which makes it possible later to use the developed apparatus for formalization of data structures and the language and program means available in the well-known technologies.

The present article is a survey or results obtained within the framework of structural programming using the multilevel structural program design technique, an important direction in theoretical and system programming. In particular we will present results that were given at the all-Union seminar "Structural Programming in Systems of Algorithmic Algebras — Kiev-81."[2] (Works [54, 69, 23] were also devoted to development of the mathematical foundations of this line of study.)

Section 1 presents the conception of structural design grammars, which is the basis of the multilevel structural program design method. The article proposes a classification of methods of program development according to ascending, descending, and mixed strategies. The problem of multilevel partial verification of programs, their transformation, and documentation in the process of development and further accompaniment is formulated on the basis of the structural design grammars apparatus. The multilevel structural level program design method is illustrated with a series of programs being designed.

Section 2 is devoted to development of the tools of structural programming by the multilevel structural level design technique. The linguistic and program means which are its foundation are presented. We review the problem of multilevel optimizing translation within which the MUL'TIPROTSESSIST system is being developed. This system is oriented to automation of structural parallel programming by the multilevel structural program design technique.

Results from application of the multilevel structural program design technique to problems of symbolic processing are given in section 3. There is a description of the process of structural design of the PARTRAN (multilevel parallel conveyor-type translator) system and components of minicomputer and microcomputer software.

Section 4 gives an abstract model of a homogeneous computing system [29, 30, 39]. Formal means of structural design of parallel programs based on the apparatus of modified SAA are developed. The article gives results from designing components of homogeneous computing systems software and multimachine complexes. The problem of automating the development of simulation modeling systems for communications networks is reviewed.

Section 5 is devoted to results of structural designing in automated control system problems. The architecture of the TEKHNOLOG system is reviewed. This system is oriented to automating the process of technological preparation for production applying the multilevel structural program design technique. There is a formal description of the process of structural design of programs to process large files within the framework of a typical automated control system problem. Prospects

---

[2]See "Scientific Information" section of this issue of the journal.

for further development of applied structural programming by the multilevel structural program design method are outlined. Figure 1 below illustrates the interrelationship of the theoretical and applied aspects of structural programming by the multilevel structural program design method.

The results presented in sections 1-3 and 5 were obtained under direction of the authors of this survey.

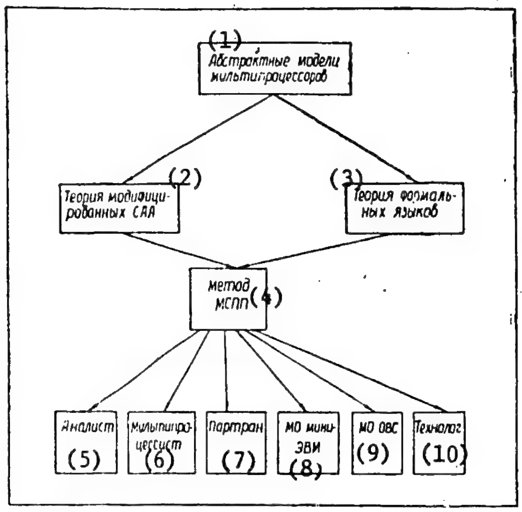


Figure 1. Multilevel Structural Program Design and Its Applications.

Key: (1) Abstract Models of Multiprocessors;
(2) Theory of Modified SAA;
(3) Theory of Formal Languages;
(4) Multilevel Structural Program Design Method;
(5) ANALIST;
(6) MUL'TIPROTSESSIST;
(7) PARTRAN;
(8) Minicomputer Software;
(9) Software of Homogeneous Computing Systems;
(10) TEKHNOLOG.

This approach to setting up subsystems for operational accounting for production was successfully used in setting up a subsystem of the automated control system of the Mikhaylovsk Mining and Concentrating Combine. The subsystem was formulated independently by the structural and nonstructural approaches (see Table 3 below). In addition, it should be noted that the structural approach made it

Table 3.

Feature of Development	Structural Approach	Nonstructural Approach
Total Development Time	91 Days	142 Days
Development of Algorithms and Programs	24 Days	52 Days
Full-Scale Debugging	5 Hours	12 Hours
Volume of Complex	560 Operations	753 Operations

possible to obtain understandable programs that are easily modified and convenient to accompany.

In conclusion we will observe that as applied structural programming by the multilevel structural program design technique develops further it is contemplated that the following will be worked out:

- apparatus for context control of output to structural design grammars to insure a flexible interface between the program modules under development and to see that efficient decisions are made in the process of their structural design;

- system of multilevel parallel verification and transformation of the circuits of the structured programs on the basis of further development of the ANALIST package;

- multilevel interactive optimizing translation-type system from the family of SAA-chart languages based on further development of the MUL'TIPROTSESSIST system;

- the PARTRAN system, components of homogeneous computing systems, minicomputers and microcomputers, and the TEKHNOLOG system in which the MUL'TIPROTSESSIST system will be used as a tool;

- formal apparatus for describing the functioning of a collective of developers in the process of compiling large program complexes by the multilevel structural program design method;

- course of programmed teaching in the multilevel structural program design method based on the STOK system.

FOOTNOTES

23. V. M. Glushkov, G. Ye. Tseytlin, and Ye. L. Yushchenko, "Problems of Analysis and Synthesis of Structured Parallel Programs," KIBERNETIKA, 1981, No 3, pp 1-16.

54. G. Ye. Tseytlin, "The Problem of Identity Transformations of the Circuits of Structured Programs with Closed Logical Conditions, parts 1-3," KIBERNETIKA, 1978, No 3, pp 50-57; No 4, pp 10-18; 1979, No 5, pp 44-51.

69. G. Ye. Tseytlin, "Aschematics of Structural Parallel Programming and Its Applications," LECTURE NOTES ON COMPUTER SCIENCE, 1979, No 74, pp 474-481.

11,176
CSO: 1863/16

UDC 519.8

INVESTIGATION OF METHODS OF SOLVING OPTIMIZATION PROBLEMS AND THEIR APPLICATIONS

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 89-113

[Article by V. S. Mikhalevich, I. V. Sergiyenko and N. Z. Shor]

[Excerpts] Making the most economical and sound decisions is an essential requirement of practical planning and management in the national economy. The broad introduction of computer hardware in recent years has demonstrated that it is not possible to achieve high efficiency and raise labor productivity if we limit ourselves to solving information problems only, without optimizing the production of output or the indicators of its economic efficiency. For this reason, the role of methods and algorithms for solving optimization problems is growing in the software of automated systems at different levels and for different purposes.

The problem area related to the methodology of making optimal decisions, investigating the properties of complex extremal problems, constructing efficient computing algorithms for optimization and appropriate software, and using these facilities to solve complex planning, design, and management problems has been developed at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences from the first days of its existence [50, 58]. In recent years significant results have been obtained in the fields of sequential methods of decision-making in the investigation of necessary and sufficient extremum conditions in complex optimization problems, constructing efficient algorithms for solving problems of nonlinear programming and rough optimization problems, the development of decomposition schemes, stochastic programming, and solving discrete and discrete-continuous optimization problems and multicriterion problems. This article briefly presents the principal results received by the research collective of the Institute of Cybernetics of the Ukrainian SSR in these fields of study, including the main applications and development of packages of applied programs.

Theoretical research in the area of construction of automated program facilities, including packages of applied programs, began at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences in the late 1950's, after the publication of work [11]. In a number of subsequent works V. M. Glushkov and his students elaborated the basic ideas presented in [11], including their application to

the formulation of packages of programs designed for solving various classes of optimization problems. This led to the development of a number of elaborate interactive packages of applied programs, for example DISPRO. It should be noted that in some cases the interactive facilities of the packages of applied programs become very effective tools for solving and investigating optimization problems [12], especially when it is necessary beforehand to study mathematical models of the problems and trace the impact of change in particular parameters in the model of the problem and the solution method being used on the optimization process. The interactive mode of solving complex optimization problems makes it possible, in a certain sense, not only to save time searching for the solution, but also to take a new approach to the procedure for finding it.

Projects on optimization methods that have been done at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences have been highly praised in our country and abroad. However, a great deal remains to be done in order that the apparatus for selecting optimal decisions which has been developed may become an effective means for solving complex economic planning and design problems. V. M. Glushkov's article [15] points out a new line of work in systems optimization, related to overcoming the limitations of the statements of the optimization problems and means of solving them. This approach envisions a hierarchy of optimization models and active participation by specialists who during the process of reaching optimal decisions adjust both the relative importance of the various criteria and the boundaries of the permissible domain of solutions. The classical statements of extremal problems occur as local sub-problems in different stages of systems optimization. The ideas expressed by V. M. Glushkov were carried out, specifically, in the DISPLAN system [12].

We hope that the scientific and program background work of the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences in solving various classes of optimization problems will make it possible to formulate new software for solving important new practical problems of optimal planning, management, and design quickly on the basis of the systems approach to decision-making.

One of the versions of the OGSRP [Generalized Gradient Release with Elongation of Space in the Direction of the Gradient], which was independently worked out in works [107] and [110] has become widely known under the name "ellipsoids method." This method can be considered a particular case of a gradient-type algorithm with elongation of space in the direction of the gradient. On the other hand, it has the following geometric interpretation: suppose the minimum of concave function $f(x)$, $x \in E_n$ is localized in an n-dimensional sphere of radius R with center at point x_0. We will compute the subgradient $g_f(x_0)$ and construct the hyperplane $P = \{x: (g_f)x_0), x-x_0) = 0\}$. This hyperplane severs the hemisphere $\{x: \| x-x_0 \| \leqslant R; (g_f(x_0), x-x_0) \leqslant 0\}$, in which the minimum is localized. We will describe the ellipsoid of minimum volume around the hemisphere. Based on considerations of symmetry it is easy to see that its center $\bar{x}_1$ is located on arc $x_0 - hg_f(x_0)$, $h = 0$ and the ellipsoid is flattened (oblate) in the direction $g_f(x_0)$. A simple calculation shows that

$$\bar{x}_1 = x_0 - \frac{R}{n+1} \frac{g_f(x_0)}{\| g_f(x_0) \|},$$

and the coefficient of "flatness" is equal to $\sqrt{\frac{n-1}{n+1}}$. . After employing the

operation of elongating the space with elongation coefficient $\alpha=\sqrt{\frac{n+1}{n-1}}$, in the elongated space we obtain a sphere with center at point y_1, an image of point x_1, and with a radius $R_1=\frac{R_n}{\sqrt{n^2-1}}$, in which the image of the minimum function f(x) is localized. The process in subsequent iterations goes exactly the same as in the first iteration.

In this interpretation the algorithm of ellipsoids can be viewed as a modification of the center of gravity method [110] in which the ellipsoid described is used to make the area obtained after severing symmetrical. The algorithm is easily generalized to solve the general problem of concave programming.

The ellipsoids method converges according to the functional at the rate of a geometric progression whose denominator

$$g_n \approx 1-\frac{1}{2n^2}$$

depends only on the dimensionality of the space. This remarkable property was employed in works [92, 43] in the construction of very interesting, from the standpoint of complexity theory, algorithms for exact solutions to problems of linear and quadratic programming with whole-number coefficients that work for a polynomial time in relation to the bit volume of the problem's raw data. These works caused a sensation abroad and the situation was exacerbated by unfounded hopes that the ellipsoids method in its primitive form would be competitive with the simplex method. However, elementary analysis of the formula

$$g_n \approx 1-\frac{1}{2n^2}$$

shows that even where n = 100 it takes 46,000 iterations to improve precision by one decimal order.

After the publication of work [92] between September 1979 and May 1980 46 articles appeared devoted to different refinements of the ellipsoids method (see bibliographic work [119]). It should be observed that most of these refinements were contained in works done earlier at the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences [10, 109]. As shown in [110, 62], the best algorithm for the class of n-dimensional problems of concave programming insures an error by the functional and constraints on the order of ε and should make $O(n \log 1/\varepsilon)$ iterations connected with computing the functions and subgradients at one point. The center of gravity method [110] which requires the same number of iterations by order is practically inapplicable because the complexity of one iteration increases very rapidly with a number of iterations where $n \geqslant 3$. To receive the same level of precision the ellipsoids method requires $O(n^2 \log 1/\varepsilon)$ iterations. Unfortunately, at the present time there is no algorithm which, while requiring $o(n^2)$ supplementary operations (in addition to calculation of the function and gradient) at each iteration, can guarantee in a theoretically substantiated manner the relative error of the order ε for $O(n^\alpha \log 1/\varepsilon)$ iterations where $\alpha<2$. However, as long ago as 1970 a method was proposed [105, 108] which in complexity is close to the OGSRP method and insures in practice a relative error on the order of ε for $O(n \log 1/\varepsilon)$ steps, which was confirmed by the results of solving a large number of test and practical problems [108, 61], although no theoretical substantiation of the rate of convergence of this method has been done yet. We are speaking of the class of

gradient-type algorithms with elongation of space in the direction of the difference of two consecutive gradients (in short form, r-algorithms). An interation of an r-algorithm has the following form. After k steps we obtain: x_h is approximation k, the k approximation; h_k, the symmetrical, positively defined matrix of transformation of the metric of the space; and, $g_f(x_{k-1})$, the subgradient at the preceding point.

In iteration (k+1) we compute:

$$1)\ g_f(x_k);$$

$$2)\ r_k = g_f(x_{k-1}) - g_f(x_k);$$

$$3)\ H_{k+1} = H_k - (1-\beta_k^2)\frac{H_k r_k r_k^T H_k}{(H_k r_k, r_k)};$$

$$4)\ x_{k+1} = x_k + h_k \frac{H_k g_f(x_k)}{\sqrt{(H_k g(x_k), g(x_k))}}.$$

In this case, as in the OGSRP algorithm, β_k, $0 < \beta_k < 1$ is a coefficient inverse to the coefficient α_k of elongation of the space at step k; h_k is the step multiplier. The different modifications of the r-algorithm are linked to a concrete choice of sequence $\{\alpha_k\}$ and the method of regulation $\{h_k\}$ [38, 105, 108]. The family of r-algorithms proved a very effective means of solving problems of ravine-type smooth and rough optimization. The very first experiments, conducted with test problems [105], showed that the r-algorithm is practically as good as the well-known quasinewtonian and conjugated direction algorithms for minimizing smooth ravine-type functions. This algorithm demonstrated high efficiency in solving minimax problems [108]. Combined with the methods of decomposition and rough penalty functions it was used successfully to solve problems of large dimensionality for sectorial current and medium-range planning in civil aviation [108], the gas industry [61], ferrous metallurgy, the radio industry, material-technical supply, and for optimization of design elements and determining parameters of location by processing geophysical data [108]. Thus, when planning the loading of USSR pipe rolling mills and the distribution of customer orders (the work was done by the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences together with the All-Union Scientific Research Institute of the Pipe Industry of the USSR Ministry of Ferrous Metallurgy) the r-algorithm was used with a decomposition scheme to solve linear programming problems with 300,000 variables and 20,000 constraints; with a number of iterations on the order of 600 a relative precision based on the functional of 10^{-6} was achieved. The advantage of this algorithm over other methods for solving complex problems of rough optimization was also confirmed by studies made at the International Institute of Applied System Analysis (IIASA) [116]. It should be noted that r-algorithms are used to find the local minimum of nonconcave linear programming problems, in finding estimates of the functional by the "branches and boundaries" method, and in other places. At the present time there is lively debate on the question of the practical competitiveness of the new iterative algorithms compared to procedures based on the simplex method for solving large-dimensionality linear programming problems. In our opinion, algorithms of the r-algorithm type have a great future in this respect, especially for solving structured linear programming problems.

FOOTNOTES

10. V. I. Gershovich, "A Method of Truncation That Uses Linear Transformations of Space," in "Teoriya Optimal'nykh Resheniy" [Optimal Decision Theory], Kiev, 1979, pp 15-23.

11. V. M. Glushkov, "One Method of Automating Programming," in "Problemy Kibernetiki" [Problems of Cybernetics], Moscow, 1959, pp 181-184.

12. V. M. Glushkov, and G. B. Oleyarsh, "The DISPLAN Interactive Planning System," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY 1976, No 4, pp 123-124.

15. V. M. Glushkov, "System Optimization," KIBERNETIKA 1980, No 5, pp 89-90.

43. M. K. Kozlov, S. P. Tarasov, and L. G. Khachiyan, "Polynomial Solubility of Concave Quadratic Programming," DOKL. AN SSSR, 1979, 248, No 5, pp 1049-1051.

50. V. S. Mikhalevich, "Operations Research and Solving Multivariant Problems," in "Tr. Ekon. Mat. Seminara/ Lab. Ekon.-Mat. Metodov AN SSSR" [Works of the Mathematical Economics Seminar/ Laboratory of Mathematical Economic Methods of the USSR Academy of Sciences], Moscow, 1962, Vyp 1, pp 1-14.

58. V. S. Mikhalevich, Yu. M. Yermol'yev, V. V. Shkurba, and N. Z. Shor, "Complex Systems and Solving Extremal Problems," KIBERNETIKA 1967, No 5, pp 29-39.

61. V. S. Mikhalevich, I. V. Sergiyenko, A. I. Kuksa, and others, "Vychislitel'nyye Metody Vybora Optimal'nykh Proyektnykh Resheniy" [Computing Methods of Selecting Optimal Planning Decisions], Kiev, "Naukova Dumka", 1979, 118 pages.

62. A. S. Nemirovskiy, and D. B. Yudin, "Slozhnost' Zadach i Effiktivnost' Metodov Optimizatsii" [Problem Complexity and the Effectiveness of Optimization Methods], Moscow, "Nauka", 1979, 384 pages.

92. L. G. Khachiyan, "Polynomial Algorithm of Linear Programming," DOKL. AN SSSR, 244, No 5, pp 1093-1096.

105. N. Z. Shor, and N. G. Zhurbenko, "Minimization Technique Using the Operation of Spatial Elongation in the Direction of the Difference of Two Sequential Gradients," KIBERNETIKA 1971, No 3, pp 51-59.

107. N. Z. Shor, "The Cut-Off Method with Spatial Elongation," KIBERNETIKA 1977, No 1, pp 94-95.

108. N. Z. Shor, "Metody Minimizatsii Nedifferentsiruyemykh Funktsii i Ikh Prilozheniye" [Methods of Minimizing Nondifferentiable Functions and Their Application], Kiev, "Naukova Dumka", 1979, 200 pages.

110. D. B. Yudin, and A. S. Nemirovskiy, "Information Complexity and Efficient Methods of Solving Concave Extremal Problems," EKONOMIKA I MAT. METODY 1976, 12, Vyp 2, pp 357-369.

116. C. Lemarechal, and P. Mifflin (editors), "Nonsmooth Optimization," PROC. INT. INST. APPL. SYSTEMS ANALYS. WORKSHOP, New York, Pergamon Press, 1977, 186 pages.

119. Ph. Wolfe, "A Bibliography for Ellipsoid Algorithm. Res. Rep.," RC 8237, New York, IBM Thomas J. Watson Res. Center, 1980, 6 pages.

11,176
CSO: 1863/16

REVIEW OF NEW BOOK ON MULTIPLE PROGRAMMING TECHNIQUES

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 p 140

[Review by M. R. Shura-Bura of book "Metody Simvol'noy Mul'tiobrabotki" (Methods of Symbolic Multiple Processing) by V. M. Glushkov, G. Ye. Tseytlin and Ye. L. Yushchenko, "Naukova Dumka", Kiev, 1980, 242 pages]

[Text] Contemporary computer technology is entering the age of highly productive multiprocessor computing systems. It can be expected that highly productive multiprocessor systems will greatly intensify the development of the machine intellect just as collective labor promoted the shaping of human reason. The level of the machine intellect is largely determined by the structure and quality of computer system software.

The monograph under review, by prominent Soviet scientists V. M. Glushkov, G. Ye. Tseytlin, and Ye. L. Yushchenko, is devoted to timely problems of theoretical and systems parallel programming that are arising in connection with the designing of language processors for multiprocessor computing systems. The overall structure of the book comprises an introduction, six chapters arranged in two sections, and an extensive bibliography.

The introduction is methodological in nature and contains a dialectical analysis of research on the hardware and software of computer systems. It emphasizes the necessity of formalizing the process of designing programs on the basis of a synthesis of algebraic apparatus and theoretical linguistic apparatus in connection with solving timely problems of theoretical and systems programming.

Section 1 consists of three chapters. It is devoted to the use of formal models of language multiprocessors. Significant attention is given to the systems of algorithmic algebra proposed as far back as 1965 by V. M. Glushkov which anticipated the conception of structural programming, which has become widespread recently. The section presents basic results on the schematology of structural parallel programming, which is based on the theory of modified systems of algorithmic algebras oriented to formalization of the multiple processing process.

There is an investigation of grammatical and automat models of languages. Within the framework of these models the mechanisms of parallel removability of three types, descending, ascending, and mixed, are developed. An apparatus is developed for synchronous parallel matrix grammars, in whose terms the controlled parallel descending removal is carried out.

Section 2, which contains chapters 4-6, is devoted to the problems of analysis and synthesis of language multiprocessors. It investigates the problem of parallel analysis on the basis of the grammatical and automat language models considered in Section 1. The section proposes two-dimensional and three-dimensional parallel matrix algorithms by means of which the linearity of the upper estimate of the time complexity of parallel syntactical analysis of context-less languages on homogeneous structures is established. These algorithms are formalized in terms of multidimensional periodically determined transformations that carry out the strategy of synchronous multiple processing.

A separate chapter is devoted to the theory of parametric models of language multiprocessors oriented to bilateral analysis, which combines descending and ascending symbolic multiple processing of programs from opposite ends. The mechanisms of context control of non-impasse bilateral analysis, on which the functioning of the corresponding parallel analyzers is based, are developed. The Ginsburg problem, consisting of a characterization of the class of unique context-free languages, is solved in terms of parametric grammars. Parametric models of language multiprocessors oriented to multilevel analysis, which is a natural generalization of bilateral multiple processing, are developed.

The authors do not restrict themselves to purely mathematical investigations. They devote adequate attention to applications of these results in connection with the designing of parallel programming systems for highly productive multiprocessor computing systems. They formulate the basic principles of the structural and program organization of such systems and give a general description of the well-known domestic projects: homogeneous computing systems, recursive machines, and the MARS [Intersectorial Automated Republic Scientific-Technical Information System] system. The book proposes an architecture for a multilayer parallel conveyor-type translator and gives structural definitions in terms of modified systems of algorithmic algebras and particular blocks. The apparatus of modified systems of algorithmic algebras is the basis of the MUL'TIPROTSESSIST system, which is an instrument for structural parallel programming.

An unquestioned strongpoint of the monograph is the completeness of the presentation, which is achieved by consistently detailing material using analogies with the conception of descending program design. The linking elements here are circuits for bilateral conveyor and carousel multiple processing whose concrete forms lead to systems processes that arise during parallel translation, supervision, designing parallel programs, and solving various other important problems of contemporary programming.

It should be noted that the authors sometimes are too brief in pointing to possible generalizations from the concepts they have introduced. In their

consideration of certain abstract concepts it would have been timely to point out the constraints that are essential for practical realization. In the process of describing multilayer analysis [Section 5.3] they do not explicitly point out the parenthetical character of the structure of constraints which insure decomposition of the program under analysis into components that can be put through multiple processing.

This monograph by V. M. Glushkov, G. Ye. Tseytlin, and Ye. L. Yushchenko is one of the first in world literature devoted to timely problems of theoretical and systems parallel programming. It gives a full treatment to the authors' original results in the theory of language multiprocessors and its applications and to domestic and foreign research touching this area (this work is dispersed in many journals and anthologies). The book is written in understandable language and has a great deal of illustrative material. There is no question that it will be useful to a broad range of mathematicians and engineers working in the field of theoretical and systems programming, in particular on designing the structure and program software of highly productive multiprocessor computing systems. It will also be useful to graduate and upper division college students specializing in the corresponding field.

11,176
CSO: 1863/16

APPLICATIONS

COMPUTER NETWORKS IN CONTROL OF SYSTEMS DEVELOPMENT

Moscow VOPROSY KIBERNETIKI: UPRAVLENIYE RAZVITIYEM SISTEM in Russian 1979 pp 47-65

[Article by S. I. Samoylenko, Moscow]

[Excerpts] The phases of development of complex systems are considered and the main functions of control computer networks are determined. Known methods of switching are analyzed. The need to use automatically rearranged switching methods is substantiated. The principle of adaptive switching is suggested. Optimization problems that arise at individual phases of control of systems development are analyzed and some approaches to finding solutions of these problems are presented.

It follows from the results found in [1]:

1. In the case of continuous messages, the packet switching method with constant virtual channel provides better results with relatively short messages in the range of approximately 800-1,000 bits.

With longer messages, the channel switching method provides better results provided that an optimum part of the capacity of the total channel (from 2 to 8 channels with speed of 2,400 bits/s) is allocated to transmit signal information related to channel organization.

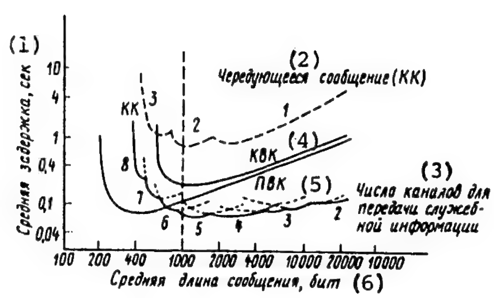


Figure 4. Dependence of Delay on Message Length with Channel Utilization Factor of 50 Percent
[Key on following page]

[Key continued from preceding page]:

1. Average delay, seconds
2. Alternate message (channel switching)
3. Number of channels for transmission of service information
4. Switched virtual channel
5. Constant virtual channel
6. Average message length, bits

2. The packet switching method provides better results at practically the entire range of message lengths with composite messages.

3. The packet switching method with constant virtual channels provides better results over the entire range of message lengths under consideration (from approximately 500 to 40,000 bits) both for continuous and for composite messages with channel utilization factor of 75 percent.

Approximately the same results were also found for channels with long propagation time, specifically for satellite lines with channel utilization factor of 50 percent and above in the range of message length from approximately 200 to 20,000 bits. It follows from the given consideration that the range of effective use of different switching methods depends both on the average length of the messages to be transmitted and on a number of other factors, specifically, on the load of the network at a given moment, propagation time in the channel and so on.

An estimate of the cost of transmission with different switching methods shows that the most economical method of transmission is achieved in packet-switching networks for short messages and in channel-switching networks for long messages. The most economical method of transmission under real conditions with mixed traffic is the integrated method in which channel switching and packet switching are combined [8-11].

Thus, specifically, the method of combining channel and packet switching on the basis of using information frames consisting of N packets of fixed length in the transmission channel was proposed in [10]. Part of the frame, consisting of N_1 packets, is used in the channel switching mode to form N_1 channels with time packing while the remaining part of length $N - N_1$ packets is used in the packet switching mode. The value of N_1 can be varied.

This system permits one to combine channel and packet switching but specific deficiencies are inherent to it. One can include the following among them:

1. Assignment of packets for an entire system of fixed length leads to underutilization of the channel capacity if the packets actually transmitted have a length shorter than that given.

2. The intervals free of transmission are not utilized within the messages upon transmission in the channel switching mode of composite messages, which reduces the utilization factor of capacity.

Adaptive Switching

It follows from the given consideration that the effectiveness of using channel capacity with different methods of communication switching and consequently the cost of transmission are considerably dependent on the nature of the messages to be transmitted.

However, the precise value of the message parameters on which depends the effectiveness of the switching method is unknown not only during design of the network but during operation of it as well and because of this, clear selection of the optimum switching method for the system being design is an essentially unsolvable problem.

The only possibility of providing high efficiency of the network is to use adaptive principles of switching in which different switching methods can be used in the network and distribution of the capacities among them is adaptive, depending on the state of the network at a given moment.

Let us consider some possible approaches to construction of computer networks with adaptive switching. An adaptive switching method is outlined below that combines the principles of packet and channel switching and that permits distribution of channel capacity for use in the channel or packet switching mode as a function of the needs occurring at a given moment. This method permits one to utilize pauses in composite messages.

The essence of the method is as follows.

Let us assume that there is a wideband channel with capacity of R_T characters per second, which can be subdivided into N channels made available to users with transmission rate of $R_P = R_T/N$ between two adjacent switching terminals. To provide adaptive switching, transmission through the channel is carried out by synchronous cycles by N characters in each, which we shall call frames. Each position of the frame can be regarded as an individual channel with rate R_P with time packing in the common channel.

The problem consists in using the frame in which the following capabilities would be provided:

1. Combination of channel and packet switching.

2. Adaptive redistribution of the total channel capacity between channels with packet and channel switching.

3. Use of transmission-free intervals in composite messages transmitted in switched channels for message transmission.

4. Transmission of packets of arbitrary length (within established limits).

5. Provision of the capability of transmitting all types of information: operational, dialogue and background.

Let us assume that it is sufficient to realize transmission in the KK [Channel switching] and KP [Packet switching] mode with two levels of priority to provide the capability of transmitting all the types of information enumerated in paragraph 5. Thus, specifically, operational information with very rigid requirements on delay can be transmitted over switched channels secured to users for the entire period of their interaction. The remaining part of the operational information can be transmitted at a higher level of priorities in the KK or KP mode depending on message length: long messages are transmitted in the KK mode and short messages are transmitted in the KP mode.

Dialogue information can usually be transmitted in the KP mode at higher or lower level of priority depending on the restrictions on delay time and also for some conditions with very rigid allowances for delay in the KK mode with the channel secured for the time of the communications session.

Background information consisting of messages of long length is transmitted in the channel switching mode with lower level of priority. Consequently, the switching terminal should provide the following capabilities:

1. Make available channels in the KK mode with two levels of priority. At the higher level, the available channel cannot be disconnected by another request and at the lower level the channel can be disconnected with preliminary notification of the information source.

2. Make available capabilities of transmission in the KP mode with two levels of priority. The difference of the levels of priority consists in different values of the permissible delay.

3. Load the free intervals in composite messages transmitted in the KK mode by transmission of messages in the KP mode.

4. Operationally redistribute the capacity of the transmission channel between messages transmitted in the KK and KP modes.

The overall principle of designing a switching system that realizes the indicated problems is shown in Figure 5. As already noted, the common transmission channel is used to combine N channels in the time packing mode. The switchboards K on the receiving and transmitting sides operate synchronously with the timing frequency equal to the rate of transmission in the channel. The inputs and outputs 1, 2, ..., ..., N are used to transmit messages in the KK mode. The input and output N + 1 are used to transmit messages in the KP mode.

The mode of using each channel depends on the position of the switches P_i, i = = 1, 2, ..., N switched synchronously on the transmitting and receiving sides. If switches P_i are located in the upper position, then the i-th channel is used in the KK mode, i.e., it is secured in a given section for organization of communications between users. If P_i is in the lower position, then the i-th channel is used to transmit messages coming from VKh_{N+1} in the KP mode. Thus, by changing the position of P_i, one can redistribute the channel capacity between messages transmitted in the KK and KP modes. As many characters as switches that were in the lower position at a given moment can be transmitted in the KP mode from the message source during a single operating cycle of the switch K.

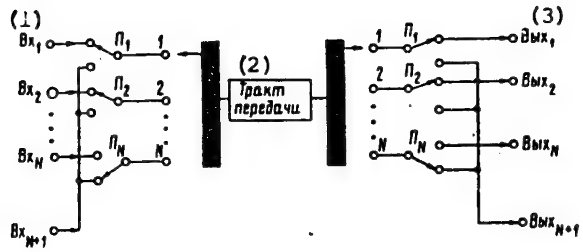


Figure 5. Principle of Adaptive Switching

Key:
1. Input
2. Transmission channel
3. Output

The following data are required to control the operation of the switching terminal:

1. The number of channels which is required at a given moment of time to ensure transmission of packets with delay not exceeding that permissible. Let us denote this value by N_p. This value depends on filling the buffers on a given time interval.

2. Data on requests for channels for transmission in the KK mode with higher priority. Let us denote this value by K_v.

3. Data on requests for channels to transmit background information. The number of requested channels is denoted by K_f.

4. Data on the appearance of pauses in each of the channels used in the KK mode. These data can be obtained if the information is transmitted in blocks in the KK mode with known boundaries as, for example, with use of flags provided by the HDLC protocol.

5. The maximum number of channels used in the KK mode of higher priority in case of complete load with requests for transmission of messages of higher priority in the KK and KP modes. This value is denoted by K_{max}.

6. The minimum number of channels used in the KP mode which are required to transmit service information related to organization of the switched channels N_{min}.

The control procedure in the switching terminal may then appear as follows depending on the schedule:

C1. There are no requests for KK. In this case all N channels are used in the KP mode.

C2. A large number of requests for KK in the absence of messages in the KP mode. The number of channels used in the KK mode is equal to $N - N_{min}$.

C3. A large number of requests for higher priority KK and transmission in the KP mode. In this case K_{max} channels are used in the KK mode and $N - K_{max}$ channels are used in the KP mode. Moreover, messages in the KP mode fill the pauses in messages transmitted by KP.

C4. A large volume of messages in the KP mode with number of loaded channels in the KK mode less than K_{max}, i.e., $K_V < K_{max}$. In this case K_V channels are used in the KK mode and the remaining $N - K_V$ channels are used in the KP mode. The intervals in messages are filled by packets.

C5. The number of required channels in the KK mode is $K_V < K_{max}$; the volume of messages in the KP mode is such that $N_P < N - K_V$. In this case $N - K_V - N_P$ channels can be used to transmit background information in the KK mode. The intervals in composite messages transmitted in the KK mode are filled by packets.

If the situation changes, the channels are redistributed (provided that the restrictions indicated above are fulfilled). The use of the principle of adaptive switching outlined above promises high efficiency of utilizing channels in the most diverse situations with different requirements for transmission of various types of messages. Its advantage is also that precise knowledge about the distribution of various types of messages is not required during system design.

Problems of Computer Network Design

The process of controlling systems development essentially consists of a nonspecific search for solutions of optimization problems.

Some of these problems are relatively simple, have clear resolving algorithms and can be solved automatically without human participation.

However, a considerable number of problems related to development and modification of complex systems, selection of control strategy in the operational phase of these systems and so on, are creative problems in solution of which man or collectives of people participate in one or another form. The methods of solving these problems are investigated within the problem of artificial intelligence.

The complexity of the system and the incomplete definiteness of the external medium leads to the fact that known methods of strict optimization are inapplicable for many problems both in the design and modification phases and in the operational phase of the system. Under these conditions suboptimum heuristic procedures of finding solutions that provide a saving of search time by refusing the obligatory finding of the optimum solution and going on to searching for "good" permissible solutions play the main role. Suboptimal methods of finding solutions are based in most cases on one or another form of using human experience in the search algorithms in solving problems of a given class.

The possibilities of formalizing human experience determine the degree of machine participation in finding the solution.

If man can only say "what is to be done" in a given situation, but cannot describe the rules for selecting the solutions, the main method of finding solutions is the use of a man-machine system. In these systems a man evaluates the given situation and formalizes the hypothetical solution. A machine is used to calculate the estimate of this solution and also to assist man in construction and analysis of other solutions close to that recommended by one or another criteria.

If the experience of finding solutions by man can be formalized, heuristic machine search procedures are constructed which ensure that acceptable, but not obligatory optimum solutions will be found.

It is impossible in the given article to outline in detail the principles of constructing heuristic search procedures. In this regard we limit ourselves only to references for publication where some program-realized methods of heuristic search are outlined and the results of their experimental investigation are presented.

Several approaches to construction of heuristic search algorithms, including man-machine procedures, are proposed in [12]. One of the approaches outlined in this paper, called the heuristic branching method, was realized on the example of the problem of distribution of a complex algorithm in a computer network [13, 14].

Known heuristic search procedures usually lead to a single solution which may differ significantly from the optimum solution in some cases. It is not yet possible to improve this solution with an ordinary heuristic search.

A method of eliminating this deficiency by introducing controlled ambiguity and diffusion into the search procedure was proposed in [15]. This method was program-realized and investigated on a number of problems, specifically, that of constructing a circular transmission line of the shortest length, i.e., in the travelling salesman problem [16], the problem of designing computer networks with multiterminal communications lines [17] and so on. The investigations showed that the open heuristics method permits one to improve the results found with ordinary heuristic search and leads to an optimum solution in the absence of restrictions on search time. Practical realization of this method is adequately simple.

Control of complex systems development is a complex, creative problem during solutionof which large volumes of information coming from territorially dispersed sources must be gathered and processed. Many collectives of people, specifically, research centers, can be included in the control process at individual phases of systems development. These collectives should have the opportunity of operational exchange of information and also joint use of territorially dispersed computer resources and software to ensure their effective interaction during solution of the common problem. Moreover, the research centers should have operational access to the data bases in the corresponding field of scientific and technical information which can also be territorially dispersed.

The most efficient means for solving many problems of this class is a computer network with adaptive switching, providing high flexibility, viability, the possibility of sequential buildup without changing the functional principles and a number of other positive characteristics.

The process of development control includes a continuous sequence of solutions of complex optimization problems, many of which are creative in nature. The main means of solving problems of this type are heuristic methods of finding solutions developed within the problem of artificial intelligence.

BIBLIOGRAPHY

1. Kummerle, K. and H. Rudin, "Packet and Circuit Switching: Cost/Performance Boundaries," COMPUTER NETWORKS, Vol 2, No 1, Feb 1978.
2. Closs, F., "Message Delays and Trunk Utilization in Line-Switched and Message-Switched Data Networks," Proceedings of the First USA-Japan Computer Conference, Tokyo, 1972.
3. Esterling, R. and P. Hahn, "A Comparison of Digital Data Network Switching," Proceedings of National Telecommunications Conference, NTC 75, New Orleans, 1975.
4. Itoh, K., T. Kato, O. Hashida and Y. Yoshida, "An Analysis of Traffic-Handling Capacity of Packet-Switched and Circuit-Switched Networks," Proceedings of Datacomm 73, Third Data Communications Symposium, St. Petersburg, Florida, 1973.
5. Miyahara, H., T. Hasegawa and Y. Teshigara, "A Comparative Evaluation of Switching Methods in Computer Communication Networks," Proceedings of the International Communication Conference, San Francisco, 1975.
6. Rosner, R. D., "Packet Switching and Circuit Switching: A Comparison," Proceedings of the National Telecommunication Conference, NTC 75, New Orleans, 1975.
7. Rosner, R. D. and B. Springer, "Circuit and Packet Switching. A Cost and Performance Tradeoff Study," COMPUTER NETWORKS, Vol 1, 1976.
8. Jenny, C., K. Kummerle and H. Buerge, "Network Node with Integrated Circuit/Packet-Switching Capabilities," Proceedings of the European Computing Congress on Communications Networks, London, 1975.
9. Jenny, C. and K. Kummerle, "Distributed Processing Within an Integrated Circuit/Packet-Switching Node," IEEE TRANSACTIONS ON COMMUNICATIONS, COM-24, 1976.
10. Kummerle, K., "Multiplexor Performance for Integrated Line-and-Packet Switched Traffic," Proceedings of ICCC'74, Stockholm, 1974.
11. Port, E., K. Kummerle, H. Rudin, C. Jenny and P. Zafiropulo, "A Network Architecture for the Integration of Circuit and Packet Switching," Proceedings of ICCC'76, Toronto, 1976.

12. Samoylenko, S. I., "Algorithms for Purposeful Stochastic Search," Report at Third International Conference on Artificial Intelligence, Stanford, United States, No 111-74, VINITI, 1973.

13. Pashayev, I. S., "Some Applications of the Heuristic Branching Algorithm," VINITI, No 7758-73.

14. Ambarov, Ye. V., "Approaches to Vector Optimization in a Computer Network," in "Kodirovaniye i peredacha informatsii v vychislitel'nykh setyakh" [Coding and Transmission of Information in Computer Networks], Moscow, 1977.

15. Samoylenko, S. I., "Diffusion Heuristics," Report at Fourth International Conference on Artificial Intelligence, Tbilisi, No 18-77, VINIT, 1975.

16. Samoylenko, S. I., "Suboptimum Programming," in "Semiotika i informatika" [Semiotics and Informatics], No 8, Moscow, 1977.

17. Samoylenko, S. I., "Applications of Fuzzy Heuristic Techniques to Computer Network Design," Proceedings of Fifth International Joint Conference on Artificial Intelligence, Cambridge, USA, 1977.

6521
CSO: 1863/5

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000500010023-0

UDC 669.1 : 658.65.011.56

AUTOMATED DATA PROCESSING IN FERROUS METALLURGY

Moscow AVTOMATIZIROVANNAYA OBRABOTKA INFORMATSII V CHERNOY METALLURGII in Russian 1981 (signed to press 18 Mar 81) pp 2-5, 198-199

[Annotation, foreword and table of contents from book "Automated Data Processing in Ferrous Metallurgy", by Engel's Yakovlevich Aganson and Georgiy Prokhorovich Drigval', Izdatel'stvo "Metallurgiya", 1630 copies, 200 pages]

[Text] The functions and structure of automated data processing systems (SAOI's) are discussed, including the OASU Chermet [Industrial Automated Control System for Ferrous Metallurgy] and its relationship to ASU's [automated control systems] of other control levels, as well as the methodology of the organization of the development, entry into service and use of SAOI's. Examples are given of advanced know-how in the creation and use of SAOI's at enterprises and in organizations of the industry.

This book is intended for engineering and technical personnel in the ferrous metallurgy industry. It can be helpful to students in the system for improving skills and to specialists involved in creating and using various automation systems, as well as to VUZ students.

Foreword

Ferrous metallurgy is one of the country's leading industrial sectors. In the major machine building industries the percentage of ferrous metals equals more than 96 percent of the total consumption of metal products.

Ferrous metallurgy's share of industrial production fixed capital has reached 11 percent of the country's industrial capital. The industry's enterprises consume more than 15 percent of the fuel produced by the country and about 10 percent of the electric power. The organization of the best utilization of this capital, fuel and electric power is one of the major national economic problems whose solution will make it possible to strengthen considerably the country's economic power.

Ferrous metallurgy is a complex industry which mines iron ore and other minerals and produces pig iron and steel (more than 1800 types) and rolled products. Also, every year the production of new more economical types of steel and rolled shapes (as many as 200) is mastered and specifications for the quality and delivery deadlines for metal are heightened.

Satisfaction of the ever heightened requirements for product quality and the industry's work with an increase in total output and strengthening of the specialization of enterprises and their cooperation have urgently necessitated improvement of control in the industry [1]. The 24th, 25th and 26th CPSU congresses specified the creation of automated data processing systems (SAOI's) as an important direction for this improvement.

At the industry's enterprises much attention has always been paid to the improvement of production control. The first centralized supervisory control system appeared in the 30's and in the 50's systems which automate the performance of individual technological operations (location and spacing of sheet metal blanks, control of melting and the like). The creation of ASU's for technological processes, shops, metallurgical production processes, enterprises and the industry was begun in the Eighth Five-Year Plan period, and in the Ninth Five-Year Plan period the development of integrated control systems which make it possible to improve considerably the effectiveness of control.

In the industry ASU's for various purposes have been created and are functioning successfully, questions relating to the theory and practical application of which are discussed systematically at all-Union conferences, seminars and meetings, as well as at international conferences on the automation of production processes and control in ferrous metallurgy.

An analysis of the functioning of ASU's which have been introduced and of progress and the organization of work in the industry relating to the creation and use of automated data processing systems has demonstrated that favorable conditions for the creation of highly effective ASU's exist in a number of enterprises and organizations. On the other hand these enterprises and organizations have experienced certain difficulties resulting from the newness and complexity of many questions of a theoretical nature, an insufficient amount of experience in using ASU's and the lack of a series of methodological materials on solving specific problems in the development, introduction and functioning of various ASU's.

Of course the development of questions relating to the theory of ASU's has been lagging behind the need for it. This makes it possible to explain the urgency of work in which the advanced know-how of enterprises and organizations in this area is analyzed and generalized and questions are discussed relating to theory and their practical application and their being put to use. This is all the more important for ferrous metallurgy, whose enterprises are characterized by large scales and output of a random nature, by the complex structure and organization of control, by a long list of products for mass consumption, by the many varieties of alternative solutions for producing the same product from different kinds of raw material and materials with different equipment, etc.

With this book the authors hope to fill the existing gap to a certain extent. In this book some aspects of control of the industry are discussed and a demonstration is given of the need to create and of the structure of automated data processing systems at various levels of control from the viewpoint of the solution of national economic problems by the industry. Using as an example the Industrial Automated Control System for Ferrous Metallurgy (the OASU Chermet), a demonstration is given of the performance of certain functions for control of the industry's

automation systems and of the relationship between the OASU Chermet and outside systems and various ASU's included in its structure. The methodology for organizing the development, entry into service and use of automated data processing systems in ferrous metallurgy is discussed.

The authors believe that this material will be of help to developers and users of these systems in improving the organization and performance of their jobs and to the administrative management personnel of enterprises and organizations of the industry, in enabling them to understand better the features and specifics of the work of computing subdivisions. This book will acquaint personnel managers who do not have special training in this area with questions relating to the creation of automated data processing systems.

The authors wish to express their appreciation to USSR Minchermet [Ministry of Ferrous Metallurgy] PEU [Economic Planning Administration] Deputy Director Candidate of Economic Sciences V.S. Olefir, USSR Minchermet Computing and Data Processing Center Chief Mathematician G.I. Kleynerman and Chermetenergo [expansion unknown] Administration Director V.I. Petrikeyev for the valuable advice and comments expressed by them in the preparation of this book.

CONTENTS

8831
CSO: 1863/27

UDC 669.1 : 658.65.011.56

FERROUS METALLURGY AUTOMATED DATA PROCESSING

Moscow AVTOMATIZIROVANNAYA OBRABOTKA INFORMATSII V CHERNOY METALLURGII in Russian 1981 (signed to press 18 Mar 81) pp 43-50, 61-66, 75-78, 125-129

[Ch I secs 5 and 8, ch II sec 9 and excerpt from book "Automated Data Processing in Ferrous Metallurgy", by Engel's Yakovlevich Aganson and Georgiy Prokhorovich Drigval', Izdatel'stvo "Metallurgiya", 1630 copies, 200 pages]

[Excerpts] 5. Industry System of Computing Centers

The industry system of computing centers (OSVTs) and the country's unified automated communications system employed represent the technical base of the industry's SAOI [automated data processing system]. Since the industry's SAOI is included in the Statewide Automated System (OGAS) as one of its subsystems, then the OSVTs is also part of the State System of Computing Centers (GSVTs), which together with the Statewide Data Transmission System (OGSPD) are governed by the technical base of the OGAS. This is in conformity with the departmental-industrial principle of the introduction of computer technology and of the creation of ASU's [automated control systems] for various purposes and of computing centers in the country.

The organizational structure of the GSVTs is hierarchical and has four levels. The highest level of the GSVTs is the foundation of the ASU's of statewide organizations (the USSR Gosplan, USSR Gossnab, etc.) and of industries of the national economy (including ferrous metallurgy), and its technical base is the main computing center of the OGAS. The second level of the GSVTs is the foundation of ASU's of Union republics. To the third level belong territorial computing centers (TVTs's) located in the capitals of autonomous republics and kray and oblast centers and representing the base for ASU's of these republics, krays and oblasts. The fourth level is the foundation of ASUP's [automated control systems for enterprises], ASUNII's [automated control systems for scientific research institutes], etc., and their technical base is the automation service.

The OSVTs is being created by taking into account the GSVTs structure described. The main objectives of creating the OSVTs are the following: the most intelligent introduction and use of computing and communications equipment in the industry; the furnishing with computing capacities of practically all enterprises and organizations of the industry by connecting them to the OSVTs as users; the creation of the needed information comfort for managers and scientific, engineering and technical and control personnel of any enterprise and organization by means of a capacity for information and reference services and for the performance of various kinds of

computing work; improvement of the dynamic characteristics of practically all control systems in the industry on account of the rapid transfer of any information and the ability to employ ideal mathematical methods in solving control problems of any complexity; and the reduction of costs and time for improving computing practices in enterprises and organizations of the industry.

The OSVTs is intended to make possible the gathering, storage, processing and transfer of any information in the interests of all of the network's users; to enable the interaction of various ASU's and computing centers in solving control problems; and for the performance of computing work on collective-use principles in the interests of all enterprises and organizations and of any worker in the industry.

From the main objectives and purpose of the OSVTs presented above it follows that it represents the base of the industry's SAOI and must make possible the performance of computing work with the presentation of various kinds of information to all enterprises and organizations of the industry, including the all-Union production associations of Vtorchermet [State Trust for the Procurement and Processing of Secondary Ferrous Metals], Soyuzogneupor [All-Union Association for the Production of Refractory Materials], Soyuzremont [expansion unknown], etc., which have little computing equipment of their own. The consolidated structure of the OSVTs is illustrated in fig 6.

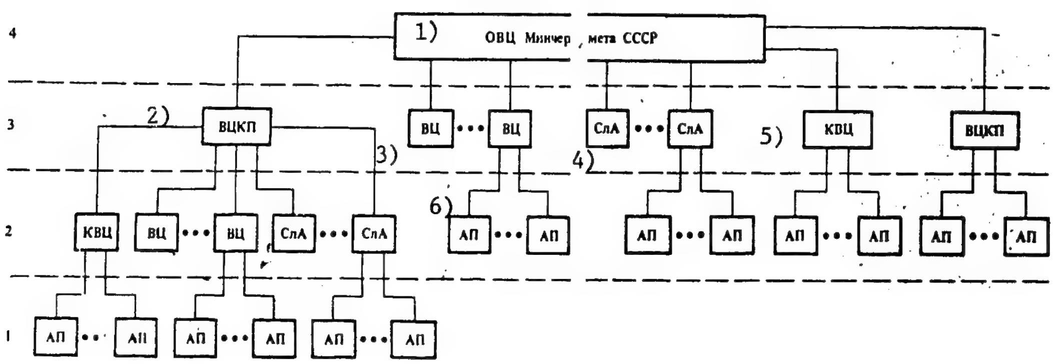


Figure 6. Consolidated Structure of Industry System of Computing Centers (The figures designate levels of the OSVTs)

Key:

1. OVTs [industrial computing center] of the USSR Ministry of Ferrous Metallurgy
2. VTsKP [collective-use computing center]
3. VTs [computing center]
4. SlA [automation service]
5. KVTs [multiple-user computing center]
6. AP [user station]

The structure of the OSVTs is four-leveled and hierarchical. At the fourth level is the industrial computing center (OVTs) of the USSR Ministry of Ferrous

Metallurgy, which coordinates and manages work relating to the creation and use of SAOI's for the highest level of control of the industry--the USSR Ministry of Ferrous Metallurgy and VPO's [all-Union production associations]. The computing center has a link with the main computing center of the GSVTs, which is used for exchanging information between the USSR Ministry of Ferrous Metallurgy and other departments and industries of the national economy.

To the third level of the OSVTs belong collective-use computing centers (VTsKP's) and multiple-user computing centers (KVTs's), which do computing work for enterprises and organizations situated nearby as well as for computing centers and automation systems. The head computing and data processing center of the Ukrainian SSR Ministry of Ferrous Metallurgy performs the functions of a large-capacity VTsKP, which does work relating to the creation and use of an SAOI for the Ukrainian SSR Ministry of Ferrous Metallurgy and coordinates the creation of a computing center network for enterprises and organizations under the jurisdiction of this ministry. The computing centers belong to enterprises and organizations in which there are no SAOI's nor even EO [economic organization] ASU's. Some of them are used as multiple-user computing centers. An automation service (SlA) is created in enterprises and organizations in case of the introduction of an SAOI or, as a minimum, an EO ASU. Some of them also perform the functions of a KVTs.

KVTs's, VTsKP's and also computing centers of GIVTs's [main computing and data processing centers], computing centers and user stations performing the functions of a KVTs must enable the access of enterprises and organizations to the information assets and hardware of the OSVTs, including the receipt of information from computing centers, automation services and user stations and the checking, shaping and storage of this information, the transfer of economic and other information to the OVTs and the receipt from the OVTs of directive instructions and other information for enterprises and organizations of the industry.

A multiple-user computing center (KVTs) is a computing subdivision of an enterprise or organization which systematically does a specific amount of computing work for a certain fee for a number of enterprises or organizations situated at a short distance from the KVTs. Sometimes a KVTs is created as an independent enterprise which does specific computing work for specific enterprises according to a schedule approved in advance by a superior organization. For example, in Krivoy Rog a KVTs has been created as an enterprise included in the head computing and data processing center of the Ukrainian SSR Ministry of Ferrous Metallurgy, for the purpose of solving problems of enterprises and organizations belonging to the Krivoy Rog region.

A collective-use computing center (VTsKP) is created as a high-output computing center with a high level of automation of the performance of all computing work. It is furnished with high-productivity computers and is designed for serving a great number of users, who transmit and obtain information through communications lines, through the mail or by special messenger. Users can be enterprises, organizations, their subdivisions or individual people who pay for the services of the VTsKP. The following services can be offered to users: the right to use the packages of applied programs belonging to the VTsKP; consultations for users relating to algorithms and programs; the formulation and development of tasks; the servicing of hardware and software at their user station; the storage of their own

files and programs on the VTsKP's magnetic media and the opportunity for access to the general-access data files of the VTsKP.

Lying at the second level of the OSVTs are KVTs's, computing centers and automation services included in the computing center network of enterprises of the Ukrainian SSR Ministry of Ferrous Metallurgy, as well as user stations (AP's) installed at small enterprises and in organizations under Union jurisdiction. In 1979 KVTs's were functioning in Krivoy Rog, Donetsk and Zhdanov. AP's make possible the performance of computing work by the facilities of the OSVTs.

By a user station is meant a subdivision of an enterprise or organization which enables the performance of their computing work by means of the OSVTs. These user stations can be small computing centers furnished with one or more computers or a subdivision equipped with various terminal equipment which enables the input/output of data on machine and other media.

An OSVTs achieves high economic efficiency if information from any point in the network is transmitted without distortion to any other point in the network, making possible the solution of control problems. In order to achieve this it is necessary to fulfill the following conditions.

1. The structure of the OSVTs and the territorial distribution of points in the network must make it possible for each user--an enterprise or organization of the industry--to perform the required computing work and to exchange information with superior control agencies.

2. The software of all users of an OSVTs must be unified and compatible in order to accomplish the one-time entry of data into the system, but the repeated use of it, to minimize the information supplied and to have 10- to 15-percent redundancy of information in order to make it possible to arrange for checking the correctness of data processing and of the restoration of data after distortions during transfer of it.

3. All users of the OSVTs must have the ability to use constantly communications channels suitable for the reliable transfer of required amounts of information over short periods of time.

4. All OSVTs users must be furnished with hardware which is compatible with each other, making possible a high level of automation of the performance of technical operations and the ability of the prolonged storage of the required amounts of information, convenience and simplicity of access to the OSVTs, and high reliability and economic efficiency in operation.

5. The OSVTs's machine processing must create a high level of automation of the performance of all technological operations of the process of processing information, controlling all hardware of the OSVTs and checking the correctness of the processing and transfer of information, as well as have high efficiency and simplicity in the utilization of all information, software and hardware resources of the OSVTs and the rapid and reliable processing of large arrays of data.

6. Users must thoroughly and precisely perform all operations and fulfill all requirements of higher levels of the OSVTs and be very disciplined and organized.

The creation of a reliable and efficient OSVTs is a very labor-intensive task requiring the completion of great amounts of research, planning, construction and other work. In particular, it is necessary to study and select the optimum structure of the OSVTs and of the territorial distribution of its elements in relation to criteria taking into account the labor intensiveness and cost of achieving the goals set, the ability to increase the number of problems which can be solved, as well as the level and extent of the automation of technological operations of the process of processing data, the enabling of reliable and efficient functioning of the OSVTs, the maximum unification of hardware employed, the required resources of the OSVTs and the like.

Let us note that cretion of the OSVTs does not assume the obligatory furnishing of all enterprises and organizations of the industry with modern computing equipment (chiefly computers) and the existence of computing subdivisions (VP's). In addition, in creating an OSVTs an endeavor is made to furnish information comfort to all personnel of enterprises and organizations on account of the redistribution of the computing capacities of enterprises and organizations for the time of the performance of computing work for them, and access to the industry data and software supply for the greatest possible number of industry personnel without considerable capital investment for the acquisition of computing equipment and the creation and maintenance of their own computing subdivisions.

Experience in the development and utilization of EO ASU's in enterprises and organizations of the industry has demonstrated that a great number of VP's and much computing and other equipment are required for the purpose of performing this work. On the other hand the mean 24-hour utilization of computers and other computing equipment is clearly inadequate--at best it has reached 18 instead of 22 hours per 24-hour period. In addition, the cost of creating user stations, multiple-user computing centers and collective-use computing centers can be reduced considerably if they are designed according to standard plans and KVTs's and VTsKP's are furnished with high-productivity computing and other equipment.

It is obvious that large and medium enterprises of the industry--metallurgical, pipe, mining and metalware--which produce large amounts of products for the national economy should have (and in part they already do) their own SAOI's and automation services. However, these enterprises, too, in addition to the ability to transfer part of their excess computing capacity, and, consequently, to compensate in part the capital expenditures, at "peak" hours can themselves enlist the additional computing capacity of other enterprises and organizations of the industry.

Work on creation of an OSVTs in the industry was begun in the 70's. At the present time the USSR Ministry of Ferrous Metallurgy on-line telegraph and telephone communications network is functioning reliably and efficiently in practically all enterprises and organizations of the industry. Every 24 hours information arrives from enterprises, chiefly through telegraph channels, on their work during the 24-hour period, on the presence of raw materials, fuel and energy resources, on the operation of major units and the like. This information is entered on punched tape of cards and is processed by computers.

Work is under way on the direct entry of information into YeS [Unified Series] computers, which will make it possible to reduce considerably the labor intensiveness

of and the time required for processing it. Development is under way on a process for the exchange of information between computers installed at the USSR Ministry of Ferrous Metallurgy Industry Computing Center and a number of large enterprises of the industry. Various kinds of methodological materials making it possible to create an OSVTs are being developed.

The proposal is to finish the basic outlines of the OSVTs in the 11th Five-Year Plan period and to complete creation of the OSVTs in the 12th.

8. Status and Direction of Development of Automation Systems in Ferrous Metallurgy

Work in the industry on the creation of SAOI's and the development of automation systems for various purposes is being performed along many lines and is pursuing the goal of the achievement of high quality and efficiency of control both by enterprises and organizations of the industry and by interindustrial associations.

This work was begun in the 30's, when keyboard and punched-card computers were used for the first time for solving very simple production problems, chiefly bookkeeping problems; the first systems for centralized supervisory control in complicated production sections and independent automatic controls for maintaining a single specific parameter (temperature, pressure and the like) at a steady level appeared. Then on the basis of analog and then analog-digital computing equipment very simple systems were created for automating the control of technological operations (most often in blast furnace production) and equipment in the advisor-expert mode. These systems were effective and contributed to the improvement of labor productivity and the quality of control, but were insufficiently reliable and convenient under changing production conditions.

Work was begun on the creation of automation systems for various purposes in the industry with the appearance of computers. The beginning of this work is associated with 1956, when for the first time in world practice the remote control by means of a computer in the advisor-expert mode of the operating cycle of a blast furnace located in Dneprodzerzhinsk was accomplished from Kiev. This experiment demonstrated the high effectiveness of digital control. Starting in 1959, at the Magnetogorsk Metallurgical Combine (MMK) and other metallurgical plants, the "Stal'" [Steel] control computer began to be used, developed by the Stal'proyekt [State All-Union Institute for the Planning of Units for Steel Foundry and Rolling Mill Production in Ferrous Metallurgy] SBVT [Special Bureau for Computer Technology] for the optimum location and spacing of sheet metal blanks and for keeping records of the production of rolled products on continuous billet mills and sheet mills. At the end of the 50's, at MMK and then at the Nizhniy Tagil Metallurgical Combine (NTMK) the development of systems for automating the on-line control of the flow of metal from steelmaking plants was begun by the Central Scientific Research Institute of Total Automation (TsNIIKA). In 1965 was assembled the first system in the USSR for controlling a section for supplying metal to the No 3 blooming mill at MKK, the "Impul's" [Pulse] system. Then systems were created for controlling the operating cycle of a blast furnace, for on-line control of steel wire production, etc. By the end of the Eighth Five-Year Plan period 37 ASU's for technological processes and production centers were introduced, about 560 different economic problems (50 percent of them accounting) and about 240 engineering problems were solved by means of computers at a number of enterprises, and the first phase of an

ASUP [automated system for controlling an enterprise] (11 functional subsystems) was introduced at the Dnepropetrovsk Tube Rolling Plant imeni Lenin and the Ingulets GOK [Mining and Concentration Combine], and 38 functional subsystems at other enterprises of the industry. This period represented the first stage in the creation of ASUP's, including the solution of particular, as a rule, relatively uncomplicated problems, which can be regarded as a preparatory stage having made it possible to gain definite know-how and, what is perhaps most important, to evaluate the effectiveness of solving individual problems by means of an ASUP.

Capital expenditures for the creation of automation systems grew considerably in the industry since the Eighth Five-Year Plan period and they have increased in each five-year plan period (table 3).

Table 3. Approximate Proportion of Capital Expenditures in the Industry for the Periods Given, Percent

Item of expenditure	1966-1970	1971-1975	1976-1980
Total capital investment for development of the industry	100.0	150.0	195.0
ASUTP's	1.8	2.33	4.46
ASUP's	0.19	0.33	1.33

The increase in capital expenditures for automation made it possible to increase the inventory of computers, including electronic computers, and to create computing subdivisions in many enterprises and organizations by means of which the front of work relating to the study and creation of automation systems for various purposes was widened (table 4).

Table 4. Growth of Number of Computers and Computing Subdivisions in Ferrous Metallurgy by Year, Number of Units

Item	1960	1965	1970	1975	1980
Computers	2	14	91	249	497
Including at enterprises	1	11	71	201	407
Computing subdivisions (computing centers, computing and data processing centers, etc.)	2	12	35	80	90
Including at enterprises	1	6	22	42	53

The second phase of the ASUP at the Dnepropetrovsk Tube Rolling Plant imeni Lenin and the Ingulets GOK, the first phase of an OASU in the USSR Ministry of Ferrous Metallurgy and Ukrainian SSR Ministry of Ferrous Metallurgy, 29 ASUP's and ASUPI's [automated systems for controlling a design institute], 70 ASUTP's and 45 computing subdivisions were put into service in the Ninth Five-Year Plan period. In 1975 108 functional subsystems were in operation and 2476 problems were solved by means

of computers, including 72 optimization, 450 production operation analysis, 800 accounting and about 300 engineering. An industry algorithm and program bank has been created which makes it possible to put in order a systematized record of algorithms and programs developed in the industry for solving various problems and the notification of enterprises and organizations regarding the existence of algorithms and programs in the bank. Exchanging them between enterprises and organizations of the industry has helped to reduce the cost of their repeated development. Work is under way at scientific research institutes and design institutes and Chermetinformatsiya [Central Scientific Research Institute of Information and Technical and Economic Research in Ferrous Metallurgy] on the development of an SANI [system for automating scientific research work], an SAPR [design automation system] and SANTI [system for automation of scientific and technical information], respectively. A system for improving the skills of managerial personnel and leading specialists in the industry has been organized and has begun to function, which improves their skills and retrains them in the area of creating ASU's. A number of metallurgical VUZ's have begun to train and turn out engineer-economists in automated data processing, and have been training production process engineers to work under conditions of the functioning of an ASU.

At the beginning of the 10th Five-Year Plan period the process of the consolidation of enterprises in the industry was basically completed, along with the changeover to a three-component control system, which to an even greater extent has occasioned the need to introduce an SAOI and ASU's for various purposes and has created the best prerequisites for the development of efficient SAOI's. The second phase of the OASU and ASUP's will be put into service at a number of enterprises in the 10th Five-Year Plan period, along with the first phase of ASUP's at 18 enterprises, 12 department ASU's, 5 ASUPI's and ASUNII's [automated control systems for scientific research institutes], and SANTI's, several SAPR's and SANI's, 120 ASUTP's, more than 20 computing subdivisions, several KVT's and VTsKP's. The first phase of an integrated ASU (an OTASU [organizational-technological automated control system]) for the steel - rolled products complex was put into service for the first time in the industry at the Novolipetsk Metallurgical Plant (NLMZ).

As we see, work is under way in the industry along a broad front, on the creation of various ASU's for various control systems for practically all levels of control. The ASU's for various purposes introduced as of 1980 will improve substantially the level of automation in the industry and will make it possible to gain a great savings. However, in analyzing the organization of work relating to the creation of these ASU's and therefore the possibility of creating an SAOI and OSVTs, it is necessary to draw the following conclusions.

1. Research and development on questions relating to the theory and methodology of the creation and functioning of ASU's for various purposes are being carried out to an insufficient extent in the industry and work has not at all begun on the creation of an SAOI and OSVTs and there is practically no system for controlling the process of creating an SAOI and OSVTs.

2. ASU's are being created at each enterprise according to individual plans, although many tasks and subsystems which have been introduced have more similarities to one another than differences. However, their use at other enterprises or in other organizations is difficult and necessitates serious revisions because of the

various procedures employed for organizing and forming software and organizational and other kinds of support.

3. ASU's for various purposes and even individual control tasks are being developed at a number of enterprises by several organizations. Poor coordination of their work and the absence of unified methodological handbook materials on the creation of SAOI's and OSVTs's which regulate sufficiently completely the performance of each technological operation of research and development processes and which make it possible to standardize design solutions and to unify and standardize all kinds of support do not make possible a saving of resources either in the development or in the use of these systems.

4. In the ASU's and subsystems now functioning control problems have not been coordinated with one another in terms of data, in order to make possible the one-time entry of source data and its repeated use in solving various problems, which has resulted in the expenditure of labor and time for the repeated preparation of this data for input and output, in making the problems which are solved more expensive and in reducing their effectiveness.

5. The data bank is still used practically but slightly in existing ASU's and dialogue systems are lacking, which does not make it possible for administrative management and other personnel to reduce considerably the time for and labor intensiveness of developing control solutions and retrieving the required data, which reduces the dynamic characteristics of ASU's and therefore their efficiency, etc.

The shortcomings of existing ASU's and those under development could be enumerated further. They are objective in nature since practical needs have outstripped the development of theory and therefore errors and non-optimum solutions are inevitable. There are, of course, also subjective factors which depend on the level of training of and understanding on the part of managers of the value of automation systems in improving the control of an enterprise or organization.

The authors believe that a serious base has been created in the industry as well as scientific and technical potential in the field of automation, that there are many highly skilled organizers and specialists in the development of ASU's and that the level of knowledge of administrative management and other personnel in the field of ASU's and their role in the creation and use of ASU's have grown. Therefore it is apropos to formulate the following guidelines for work in the industry which are to make possible the creation of high-efficiency SAOI's in the next few years with the expenditure of a minimum of resources.

1. To create an ASU for any purpose as a component of an SAOI for an enterprise, organization, all-Union production association or industry. In order to make this possible the following must be done: It is necessary to develop and introduce for strict execution by all enterprises and organizations detailed methodological handbook materials on the creation of SAOI's regulating and guaranteeing uniformity in the performance of technological operations of research and development processes; to develop standard organizational, functional, software and hardware structures for SAOI's and specifications for various kinds of support for all ASU's which are to make possible their efficient functioning in SAOI's for an

enterprise and industry; and to unify and standardize the greatest possible number of elements of each kind of support.

2. In order to improve considerably the efficiency of the operations of an industry, all-Union production association, enterprise and organization it is necessary to create information comfort for each of their workers and to make possible the rapid retrieval of necessary information, its processing and output in a form convenient for this. For this purpose it is necessary to create an OSVTs to cover all enterprises and organizations of the industry, to furnish personnel with terminals connected to the OSVTs and to develop and employ data retrieval and dialogue systems.

3. It is necessary to make it possible for practically all enterprises and organizations of the industry to perform various kinds of computing work. For this it is necessary to organize KVTs's and VTsKP's, supplying them with high-productivity computers and other hardware, to make it possible for these KVTs's and VTsKP's to communicate with the OSVTs, to connect to them as users small nearby enterprises and organizations which do not have their own computing subdivisions, and to outfit KVTs's and VTsKP's with developers for the purpose of completing projects relating to users' problems.

4. The following are necessary in order to save money and time for the development and introduction of SAOI's: to organize the planning of ASU's for any purpose on the basis of standard design solutions; to develop standard modules on the basis of successfully functioning tasks, subsystems and ASU's; to exclude the development of ASU's and SAOI's for individual projects; to provide incentives for enterprises and organizations to use standard designing; and to create in the industry an industry bank of standard modules based on the industry bank of algorithms and programs.

The data facilities of the OASU Chermet [Industry Automated Control System for Ferrous Metallurgy] include a data bank, data media, a system of classifiers and data flows.

The data bank (fig 8) contains the following economic organization information.

1. Reporting--for enterprises and organizations of the industry--in which statistical and bookkeeping information over the course of a year comprises about $300 \cdot 10^6$ characters (50 percent of the total amount) and production operation information (5000 types) $60 \cdot 10^6$ characters (8 percent of the total amount).

2. Planning (variants of plans and approved quotas for production, transport, deliveries and the like)--for enterprises, organizations and all-Union production associations. Its annual volume equals $260 \cdot 10^6$ characters (40 percent of the total amount).

3. General-system dictionaries--for enterprises and organizations--units of measurement, of production, etc.--and local--types of equipment, allied supplies, etc.--comprising $2 \cdot 10^6$ characters.

4. Norm-setting (norms for consumption and supplies, equipment performance indicators, etc.)--for enterprises, organizations and all-Union production associations, comprising $5 \cdot 10^6$ characters (2 percent of the total amount).

5. Information-reference bank.

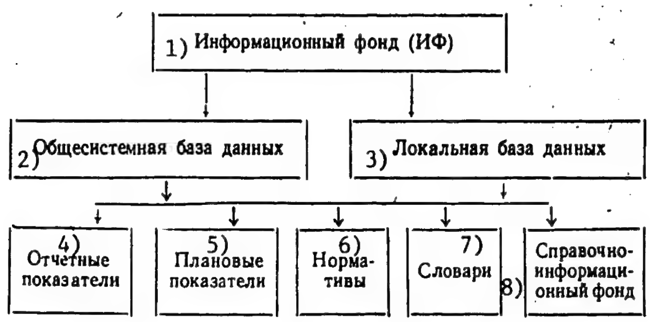


Figure 8. Structure of OASU Chermet Data Bank

Key:

1. Data bank (IF)
2. General-system data base
3. Local data base
4. Reporting figures
5. Plan figures
6. Norms
7. Dictionaries
8. Information-reference bank

The distribution of the annual amount of information (millions of characters) processed in functional subsystems of the OASU Chermet in 1978 is presented in table 6.

Table 6. Amount of Information Processed in Subsystems of the OASU Chermet, Million Characters

Short name of subsystem	Conventionally permanent information	Variable information	Total
"Longterm Planning"	2.85	0.3	3.15
"Current Planning"	6.10	240.5	246.60
"On-Line Control"	16.05	39.5	55.55
"Accounting"	2.10	249.0	251.10
"Analysis"	0.02	6.2	6.22
"Transportation"	0.30	0.6	0.90
"Supply of Materials and Equipment"	0.05	0.05	0.10
"Science and Design"	0.40	19.5	19.90
"Finances"	0.01	8.4	8.41

[Continued on following page]

"Personnel"	0.02	3.5	3.52
NTI [scientific and technical information]	0.05	0.4	0.45
Total	27.95	567.95	595.90

The OASU Chermet outputs about $400 \cdot 10^6$ characters and stores about $500 \cdot 10^6$ characters of information in the course of a year.

The basic data medium in the OASU Chermet is documents, on which about 95 percent of the information arrives. The primary documentation is represented by a set of government reporting forms--statistical (67) and bookkeeping (10)--and intradepartmental reporting--production operations (39), norm-information (66) and planning (124). Thus, information is entered into the OASU Chermet from 306 forms of documents.

In the OASU Chermet documentation has been unified by means of the "Dokument" system and the structure of reports on production operation information arriving from enterprises and organizations through communications lines in the form of teleprinter messages. This has made it possible to reduce the kinds of messages from 49 to 3, which has improved considerably the reliability and efficiency of functioning of the on-line control subsystem.

The following have been made standard in the "Dokument" system: unified presentation of the forms of documents, a unified data base, a unified technology for going through tasks (cf. fig 4 [not reproduced]), and the standard representation of information on magnetic tape.

Eleven all-Union, five industrial and 89 local classifiers are used in the OASU Chermet. Of the all-Union, classifiers are used for products; enterprises; minerals; water power resources; management documentation; sectors of the national economy; services in machine building, transportation and industry; administrative-management division and government control agencies. Industry classifiers are used for transferring production operations information, and local for solving individual problems of specific subsystems of the OASU Chermet. A great number of local classifiers hampers both the functioning of the OASU Chermet and its interfacing with other ASU's. Therefore, at the present time a gradual transition is being made to all-Union and industry classifiers.

Information flows supply functional subsystems of the OASU Chermet with the required information. When machine media are used information is transferred according to the pattern: USSR Ministry of Ferrous Metallurgy Industry Computing Center - Ukrainian SSR Ministry of Ferrous Metallurgy Main Computing and Data Processing Center - enterprise computing subdivision. Thereby the industry computing center receives a considerable part of its information directly from the computing subdivisions of enterprises. In individual cases the reception-transmission centers of enterprises are used for the transfer of data. This system of exchanging information with the utilization of computing subdivisions is promising since there is the possibility of arranging for exchange in the automatic mode directly between computers. In the transfer of information on document media a minority of it is transferred according to this system and the majority according to the following pattern: USSR Ministry of Ferrous Metallurgy - Ukrainian SSR Ministry of Ferrous

Metallurgy - enterprise. Thus, documents from enterprises first arrive at the all-Union production association or ministry and after a certain period at the industry computing center or main computing and data processing center, which results in a great delay in processing them.

The technical facilities of the OASU Chermet include the hardware complex, the data processing technology and the organizational structure of computing subdivisions which is accomplishing the creation and operation of the OASU Chermet.

The hardware complex of the OASU Chermet consists of two computing complexes (VK's): a main complex and one for processing real-time information and machine media preparation, communications and office mechanization facilities. The main VK consists of two computers, a YeS-1033 and YeS-1045, and is used for solving planning, accounting and analysis problems. The VK for processing real-time information consists of two YeS-1022 computers which are connected to communications lines by means of data transmission multiplexers.

The machine media preparation facilities are used for the transfer of information from documents. This transfer is accomplished chiefly onto magnetic tape by means of an SPD-9000 [automated data preparation system] which has made it possible to improve conditions for and to increase the labor productivity of operators.

The communications facilities include the use of leased telephone and telegraph communications channels and equipment making it possible to transfer information between the OASU Chermet and OASU Ukrchermet [OASU for Ukrainian Ferrous Metallurgy] and ASUP's. Data transmission equipment with a speed of 200 and 600-1200 bauds is being introduced. The office mechanization equipment includes duplicating equipment, equipment for forming documents, etc.

9. Questions Relating to Further Development and Improvement of the OASU Chermet

The engineering assignment for the development of the OASU Chermet and the organization of work relating to creation of its first phase was approved by decree of the USSR Ministry of Ferrous Metallurgy in 1972. The engineering assignment stipulated that the OASU Chermet will consist of 17 functional subsystems presented in table 5 [not reproduced].

This decree established the development procedure and designated the managers responsible for the creation of subsystems and hardware and software for the OASU Chermet and leading development organizations and cofulfillment organizations, and also stipulated their functions. For example, those responsible for the development of a specific control subsystem for the ministry, in addition to general management and supervision of development progress, organize introduction of the subsystem and must take direct part in the development of plans.

The leading development organization is obliged to define precisely the purpose and objectives of the subsystem, to produce an interesting formulation of objectives, to develop a methodology for functioning of the subsystem, along with plans, and to direct the work of coperformers. Such organizations have been the TsNIIChM [Central Scientific Research Institute of Ferrous Metallurgy imeni I.P. Bardin] Institute of Economics (IEChM), VNIIOchermet [All-Union Scientific Research Institute of the Organization of Production and Labor in Ferrous Metallurgy], NIIAchermet

[Scientific Research Institute of Automation of Ferrous Metallurgy], DonNIIchermet [Don Basin Scientific Research Institute of Ferrous Metallurgy] and the USSR Ministry of Ferrous Metallurgy OVTs [Industry Computing Center].

Coperformer organizations are obliged to complete a specific job by a prescribed deadline by assignment from the leading organization. Many industry scientific research institutes and planning institutes as well as institutes of the USSR Academy of Sciences and USSR Ministry of Instrument Making, Automation Equipment and Control Systems have been enlisted as coperformers.

The first phase of the OASU Chermet includes seven subsystems: "Longterm Planning," "Current Planning," "On-Line Control," "Accounting," "Analysis," "Supply of Materials and Equipment" and "Transportation."

The engineering project was completed and approved in 1973 and in 1974 a working project was developed for creation of phase one of the OASU Chermet, which was introduced in 1975 with 11 functional subsystems (222 tasks). The subsystems "Labor and Personnel," "Control of Financial Operations," "Control of Planning and Research Work," etc., were put into industrial service in addition to those called for by the plan. Tasks were introduced as their development was completed, which was conducive to improvement of control of the industry and to the revelation of shortcomings in the solution technique, formulation of the problem, the organization of operations, etc. The annual savings equaled 3.6 million rubles, basically on account of solving planning optimization problems. The capital invested was recovered in 1.8 years.

Along with this a number of shortcomings were found in organization of creation of and in plans for the OASU Chermet.

The large number of organizations which took part in the development of the OASU Chermet required great effort on the part of the industry computing center (OVTs) with regard to scientific methodological guidance and coordination of their activities. The main effort was concentrated on the development of control tasks without the prior solution of general-system problems--the hardware complex, software and other kinds of support, functional structure and especially the relationship of the OASU Chermet to ASU's of other control levels and the like. Not all organizations fulfilled their obligations; some of them presented development work not completed at the proper scientific and technical level. In addition, development directors, exhibiting caution in determining the structure of phase-one tasks, emphasized accounting problems. All this did not make possible a high level of the automation of data processing and obtainment of a large savings.

In spite of these and other shortcomings, as a result of the development of phase one of the OASU Chermet experience was acquired, a team of developers was formed at the industry computing center and within the structure of IEChM, VNIIOchermet, NIIAchermet and DonNIIchermet, relations were developed between the client for the OASU Chermet (the USSR Ministry of Ferrous Metallurgy) and developers and trust in the solutions offered was increased on both sides.

The results of the development and industrial utilization of phase one of the OASU Chermet were analyzed intensely in functional administrations of the USSR Ministry

of Ferrous Metallurgy and were discussed by the USSR Ministry of Ferrous Metallurgy NTS [Scientific and Technical Council] and decisions were made relating to the further development of the OASU Chermet on a modern basis with the employment of the latest achievements of science and technology.

The engineering assignment for further development and improvement of the OASU Chermet with the employment of standard design solutions and third-generation computers was approved by decree of the USSR Ministry of Ferrous Metallurgy in 1977. This assignment specifies the key objectives and trends of further work relating to creation of the OASU Chermet.

In order to be able to create an OASU Chermet which has a high level of automatic data processing it is advisable first to create its longterm physical base--the hardware of the OASU Chermet--to formulate an industry information base with high technical and economic characteristics and to employ modern software. And then on this basis to create a highly effective OASU Chermet and an industrial SAOI.

The hardware of the industry SAOI takes the form of a hierarchical automated data processing system the organizational and technical basis of which is KVTs's, VTsKP's, enterprise IVTs's, the computing centers of organizations, the USSR Ministry of Ferrous Metallurgy GIVTs and the OVTs. These computing subdivisions must be furnished with third- and fourth-generation computers and peripheral equipment and be interconnected by means of high-speed communications lines.

The technical and other forms of support of the OASU Chermet are discussed above in sec 2 of this chapter. It is necessary to mention that for the purpose of reducing the labor intensiveness of developing and operating the OASU Chermet and for making it possible to create an industry SAOI extensive use is made of the employment of type designs and unification and standardization facilities, e.g., the "Dokument" system, unification of communications, all-Union classifiers, equipment of the same type and the like.

These and other facilities make it possible to employ a standard data processing technology, which reduces the labor intensiveness of preparing data on account of the elimination of unnecessary redundancy and improves the reliability of information.

Compatibility of the tasks of functional subsystems is accomplished by means of the software of the OASU Chermet, which is developed by taking into account a system of rules regulating the processes of the creation of programs and the composition of the description of tasks in keeping with the engineering assignment for programming (cf. ch III). This should make possible control of a unified industry program bank and of the complex of interfaces of the industry computer network and data processing complexes using any computer in the network.

In the 10th Five-Year Plan period work was completed on further development and improvement of seven subsystems of the OASU Chermet: "Longterm Planning of Development of the Industry," "Technical and Economic Planning," "On-Line Control of Production," "Control of Transportation Work," "Statistical Reporting and Accounting," "Control of Supply of Materials and Equipment" and "Personnel Planning, Accounting and Analysis." The development of these subsystems is being carried out

by means of complexes of tasks taking into account their relationships to other tasks of this and other subsystems. This will make it possible to eliminate many shortcomings of subsystems developed earlier and to proceed to a more ideal development methodology making it possible to create ASU's with a high level of automation.

The structure of the tasks of practically all subsystems besides the last two was discussed above. The consolidated structure of the complex of tasks of these subsystems is presented below. In the "Supply of Materials and Equipment" subsystem complexes of tasks are developed relating to forecasting norms for consumption and demand in the industry of material and fuel resources and for planning of this demand (in keeping with Chermetsnab's [Ferrous Metallurgy Supply Administration] list); to the distribution and development of assets for material and fuel resources in keeping with the listing groups of Chermetsnab; to constructing balances for materials and fuel; to drawing up a combined quota for the yield of scrap and nonferrous metal waste per enterprise for all-Union production associations and the industry; to accounting for resources and controlling their realization and the flow of supplies for individual kinds of material resources (rolled metal products); and to controlling and regulating stock and controlling the flow of resources.

In the subsystem "Personnel Planning, Accounting and Analysis" complexes of tasks are being developed relating to current planning for improving the skills of managerial personnel and specialists in the industry; to keeping records of managerial and engineering and technical personnel in the industry and to analyzing their makeup and mobility; to creating an information system making it possible to form responses to requests; and to reporting on the added demand for diploma-holding specialists in the USSR Ministry of Ferrous Metallurgy.

In the 11th Five-Year Plan period it is planned to continue work on the creation of the OASU Chermet for the purpose of including to a fuller extent the personnel of the ministry and all-Union production associations in the automated data processing system, which is necessary to them for the effective performance of their functions, and for the purpose of improving the efficiency and efficacy of control of the industry's enterprises on account of presentation of the necessary production operations information.

8831
CSO: 1863/27

CONFERENCES AND EXHIBITIONS

COMPUTER DESIGN AUTOMATION CONFERENCE HELD

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 81 pp 106-107

[Report by D. I, Batishchev and V. A. Gulyayev on 9th All-Union Conference-Seminar on "Current Problems in Computer Design Automation" held in Simferopol', 4-13 May 1981]

[Text] In accordance with the plan of the Ministry of Higher and Secondary Specialized Education of the USSR and the RSFSR, the 9th All-Union Conference-Seminar on "Current Problems in Computer Design Automation" was held in Simferopol' from 4 through 13 May 1981. The conference was organized by the Moscow Institute of Electronic Machine Building, the Leningrad Institute of Electrical Engineering imeni V. I. Ul'yanov (Lenin) and the Simferopol' State University. Nine plenary sessions were conducted: "Circuit Engineering Design," "Questions of Optimization in Problems of Design," "Engineering Design," "Diagnosis of Digital Devices," "Microprocessor Systems," "Computer Network Architecture," "Computer Network Analysis," and "Automation of Design of Analog-to-Digital Converters."

The following sections were active in the conference: Computer Element Base, Mathematical Models of Transistors and Transistor Structures, Logical Design, Packages of Programs for Circuit Engineering Design, Computer Design Automation, Electronic Circuit Analysis Methods, Engineering Design of Printed and Large-Scale Integrated Circuits, Structural Synthesis, Construction of Diagnostic Tests, Design of LSI Topology, Organization of SAPR [Computer-Aided Design Systems [CAD]], Software for Microprocessor Systems, Computer Network Implementation and Modeling, Design of Microprocessor Systems, Software for CAD and Computer Networks, Mathematical Questions of Computer Design and Automated Design and Monitoring of Analog-to-Digital Converters.

In his paper, "Status and Prospects for Computer Design," P. P. Sypchuk generalized the major information on the main problems faced by developers of CAD and computers. In particular, he noted the problems of development of microprocessor systems, for example, the need for joint modeling of hardware and software, the difficulties in developing micromodels of the functional assemblies of computers, etc.

The paper, "Communications in Microprocessor Systems," was delivered in a plenary session by Ya. A. Khetagurov; in it he compared the merits of unibus and radial organizations of communication as applied to the problem of raising the throughput of microprocessor systems. Ya. A. Khetagurov is from Moscow. The paper, "Composition and Prospects for Development of the Element Base of Processors for Computers of the Next Generations," by B. N. Fayzulayev (Moscow) was also delivered in a

plenary session; in it, based on a mathematical model for forecasting, he showed the trends in reduction of delay and operation of processor base element switching, of the number of LSI circuits and printed circuit boards, the machine cycle and processor power consumption with the simultaneous sharp increase in scale of integration of the base elements and throughput. It was noted that matrix LSI circuits and microprocessors form the basis of this element base.

The paper by A. T. Yeremin (Moscow), "Problems of Automation of Design of Hardware for CAD Systems," covered the main characteristics of the sector system of design of the YeSAP [Unified System of Design Automation], implemented on the base of the YeS EVM [Unified System of Electronic Computers] hardware and intended for CAD of computer control units and logic circuits, microprogram automata and printed and modular units in the form of panels and TEZ's [standard exchange cards].

In his paper, "Modeling Malfunctions in Electronic Analog and Hybrid Circuits," A. I. Petrenko (Kiev) discussed the capabilities of circuit engineering design programs for detecting and localizing malfunctioning circuit components based on external measurements on a limited number of external leads.

K. K. Perkov (Leningrad), in his pape., "Structure and Organization of an Educational CAD System for Design of Electronic Devices," presented the main features of implementation of an educational CAD system based on the Unified System of Electronic Computers and the SM EVM [System of Small Computers] which allows easy modification of the set of application program packages in use.

The paper by B. N. Bormakov (Moscow), "Comparative Study of Domestic Programs for Circuit Engineering Design," covered the results of solving a set of test problems for the programs AROPS [Automatic Calculation of Optimal Circuit Parameters], SPROS [expansion unknown] (Moscow Aviation Institute), SPARS [expansion unknown] (Kiev Polytechnical Institute), PAUM 2 [expansion unknown] (Moscow Institute of Electronic Machine Building), PARM [expansion unknown] (Moscow Higher Engineering School imeni N. E. Bauman), ELAIS [expansion unknown] (Moscow Engineering Physics Institute); the good functional capabilities of the SPARS and SPROS programs regarding complexity of problems solved and speed were pointed out.

The ensuing discussion touched on questions of the technique of selecting and comparing programs for circuit engineering design, degree of their universality and specialization, questions of the technique of making use of CAD systems in the training process while training CAD system users, and the importance of problems of diagnosing malfunctions in complex electronic systems.

The paper by I. P. Norenkov, "Multilevel Optimization of LSI Circuits," dealt with a sequential procedure for obtaining well-grounded engineering requirements for the individual stages of parametric synthesis of devices built with LSI circuits.

In his paper, "Representation of Binary Relationships by Vector Criteria," V. V. Federov presented the decomposition approach to organization of the process of design of a complex engineering object described by differential equations and based on making use of aggregated criteria.

In his paper, "Multicriteria Problems of Topological Design of Printed Circuit Boards and Hybrid Integrated Circuits," D. I. Batishchev presented an approach to solving multicriteria problems of the design of the topology of electronic devices, and gave the results of practical studies, in particular, the results of calculations made by various algorithms for a certain class of devices.

B. A. Selyutin dealt with the problems of computer-aided design of the topology for custom LSI circuits applicable to the various technologies for manufacturing them.

In his paper, "Basic Algorithm for Tests of a Digital System and Diagnostics of Its Malfunction," V. P. Panferov proposed making use of the apparatus of tensor calculus for describing the behavior of discrete devices and extended it to solving the problem of constructing tests for checking working order.

The paper "Intelligent CAD Systems," delivered by V. A. Mishchenko, referred to the problem of automating the early stages of designing computers. The author suggested methods of constructing multifunctional automata and gave examples of hardware developed.

An approach to organizing suitable software (in particular, cross systems) for constructing control systems was presented in the paper by P. P. Sypchuk and A. D. Ivanikov, "Automation of Development of Software for Microprocessor Control Units."

In his paper, "Current Problems of Design of LSI Circuits for Digital-to-Analog and Analog-to-Digital Converters," B. G. Fedorkov presented the status of research in this field here and abroad, noted promising directions and briefly covered specific developments.

Other papers included "Architecture of Network Terminal Complexes and Automating Design of Them," by V. V. Pirogov and S. F. Gaysterov; "Adaptive Switching in Computer Networks," by S. I. Samoylenko; "Design of Specialized Program Modules for Analysis of Information Computing Networks," by S. D. Pashkevich; "Mathematical Modeling of Computer Networks with Adaptive Control," by Yu. D. Umrikhin; "Using Microprocessors in Biomedical Systems," by Ye. P. Balashov; and "Design of Algorithms for Arrangement and Layout for Design of Matrix LSI Circuits," by Zh. N. Zaytseva and V. I. Bodryagin.

Delivered in the concluding plenary session were the papers "Aspects of Development of CAD Systems for Analog-to-Digital Converters of Angular Translations" by V. G. Domrachev, and "System for Study and Monitoring of Analog-to-Digital Converters" by V. K. Shmidt.

Progress reports made in the sections were devoted to current questions of design of digital computers, in particular to design of discrete devices with the use of PLM [programmable logic arrays] and microprocessors, construction of self-synchronizing circuits, construction of application program packages; theory and technique of design of specialized digital computers for control systems, problems of achieving reliability by constructing diagnostic systems, methods of constructing tests and simulation modeling of electronic devices, principles of organization of a CAD system and its elements, algorithms and programs for engineering design, development of software for microprocessor systems and others.

The conference summed up the results of research on the problems of CAD systems for computers, resolved to further develop the most promising scientific and applied research, and promoted strengthening creative ties between representatives of VUZ science and specialists in industrial enterprises.

8545
CSO: 1863/10

8TH ALL-UNION CONFERENCE ON CONTROL PROBLEMS

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 141-142

[Article by S. F. Kozubovskiy]

[Text] The 8th All-Union Conference on Control Problems was held in Tallinn between 6 and 10 October 1980. The organizers of the conference were the Institute of Control Problems of the USSR Academy of Sciences, the USSR National Committee on Automatic Control, the Institute of Cybernetics of the Estonian SSR Academy of Sciences, and the Tallinn Polytechnic Institute.

In recent years the USSR National Committee on Automatic Control and leading scientific organizations in the country have held all-Union conferences on control problems every three years. These meetings promote exchange of scientific information in the field of control theory, its applications, and control hardware. The conferences usually precede international congresses of the International Federation of Automatic Control (IFAC) so that the most interesting reports given at the conference can be selected for the IFAC congress.

The 1st All-Union Conference on Automatic Regulation was held in Moscow in 1940. Some 100 representatives from six organizations took part in it. In 1953 the 2nd All-Union Conference on Automatic Regulation was convoked in Moscow. This conference was attended by 700 persons and 74 reports were given. The 3rd All-Union Conference on Automatic Control (Engineering Cybernetics) opened in 1965 in Odessa with 1,100 participants. At this conference, which was held on board the steamship Admiral Nakhimov, 196 reports were given. The 4th All-Union Conference (Tbilisi, 1968) had 1,200 participants and the number of reports and communications given reached 330. The 5th All-Union Conference on Control Problems was held in Moscow in 1971. Its statistics were: more than 1,000 participants from 68 cities, and 254 reports and communications. The sixth conference (Moscow, 1974) had more than 1,000 participants from 76 cities and heard 268 reports and communications. The 7th All-Union Conference (Minsk, 1977) had more than 1,000 participants from 65 cities and heard 505 reports and communications.

The 8th All-Union Conference on Control Problems was attended by more than 1,000 scientists and specialists from 70 cities in all the republics of the Soviet Union.

About 40 percent of the participants represented the USSR Academy of Sciences, the academies of sciences of the Union republics, and the higher educational institutions of the country, while about 60 percent came from sectorial scientific research institutes and production associations and enterprises. More than 2,000 reports were submitted in advance. Abstracts of 255 reports were included in the three-volume collection "8th All-Union Conference on Control Problems. Tallinn, October 1980. Abstracts of reports. Books 1-3. USSR Academy of Sciences, Estonian SSR Academy of Sciences, 1980," which was published before the conference opened. The remaining reports were published as an appendix to the collection.

The conference was opened by Academician V. A. Trapeznikov, director of the Institute of Control Problems of the USSR Academy of Sciences. Participants at the meeting devoted a minute of silence to the memory of Academician Boris Nikolayevich Petrov, vice president of the USSR Academy of Sciences and prominent Soviet scientist in the field of automatic control who died on 23 September 1980.

In his short introductory talk Academician V. A. Trapeznikov noted that the 8th All-Union Conference on Control Problems was meeting during the concluding days of the 10th Five-Year Plan and preparation of the law on the new, 11th Five-Year Plan, as well as just before the 26th CPSU Congress. The 8th All-Union Conference, he pointed out, would be an examination of scientific advances in the field of automatic control in the preceding three years.

Of the more than 2,000 reports submitted for the conference, 637 were selected for presentation. These were the reports judged to best promote scientific-technical progress and penetration into promising new fields of technology. The attention and efforts of scientific collectives today are concentrated on solving these problems and shortening the time required to put scientific advances to practical use. This necessitates a continued broadening of scientific ties with production and broad dissemination of promising new developments.

B. E. Saul', deputy chairman of the Estonian SSR Council of Ministers, gave a welcoming talk to participants at the congress on behalf of the Central Committee of the Estonian Communist Party and the Estonian SSR Council of Ministers. B. G. Tamm, academician of the Estonian SSR Academy of Sciences, rector of Tallinn Polytechnic Institute, chairman of the Estonian territorial group of the USSR National Committee on Automatic Control, and a member of the Presidium of the Estonian SSR Academy of Sciences greeted the conference on behalf of the Presidium of the Estonian SSR Academy of Sciences, the Estonian SSR Ministry of Higher and Specialized Education, the Estonian territorial group of the USSR National Committee on Automatic Control, the Institute of Cybernetics of the Estonian SSR Academy of Sciences, and Tallinn Polytechnic Institute.

The following plenary reports were given: V. A. Trapeznikov — "Some Prospects for the Development of Control Systems"; B. N. Petrov (now deceased), "Some Problems of Controlling Spacecraft"; E. A. Yakubaytis, "The Architecture of Computer Networks"; O. M. Belotserkovskiy, "The Computer Experiment and Problems of Training Personnel in the Field of Control"; M. A. Ayzerman, "Issues of Voting Theory."

In connection with the limited time (150 working hours) the section meetings were not conducted in the traditional manner. Abstracts of reports were not presented, but instead were included in a general survey report which discussed the current state and prospects for projects included in the program of the section. The remaining time was given over for discussion. This type of organization promoted broader and more vigorous exchange of information among participants at the meetings.

The conference had the following sections:

1. The Dynamics of Control Systems;
2. Methods of Functional and Structural Optimization;
3. Data Processing and Control in Stochastic Systems;
4. Machine Modeling of Control Systems;
5. Identification of Complex Systems;
6. Adaptive Control;
7. Methodological Questions of Constructing Automated Control Systems (Industrial Processes, Production Facilities, and Sectors of the Economy), and the Economic Efficiency of Automated Control Systems;
8. Methods of Processing and Representing Data Arrays in Control Systems (Image Recognition, Classification, Image Processing, Allocating Data Parameters, Preliminary Processing, and Compressing Data);
9. Control of Economic and Organizational Systems;
10. Control in Biology and Medicine;
11. Control of Moving Objects;
12. Problems of Constructing Man-Machine Complexes;
13. Control of Robots and Robot Engineering Systems, the Artificial Intellect;
14. Automating the Designing of Control Systems and Equipment;
15. Automating Scientific Research;
16. Problems of Constructing Systems and Networks for Data Processing and Control;

17. Principles of Constructing Control Hardware and Software;

18. Control of Ecological Systems;

19. Theoretical Issues of Insuring Reliability, Technical Diagnosis of Control Systems;

20. Logical Control Systems.

A total of 48 section sessions were held. In addition, the following problem reports were given: corresponding member of the USSR Academy of Sciences B. F. Lomov, "The Active Person in the Control System"; Academician of the Estonian SSR Academy of Sciences G. I. Naan, "Global Demographic Processes and Problems of Self-Regulation"; A. A. Visenkov, "Prospects for the Development of Radio Electronics"; and, P. M. Katsura, "Problems of Controlling a Large Industrial Association."

The program of the conference also included round table debates on three topics: "The Goals, Problems, and Prospects of the Science of Control" (led by academician of the Estonian SSR Academy of Sciences B. G. Tamm); "Robot Engineering in Control Problems" (led by I. M. Makarov); and, "The Role of Microelectronics in Control Systems" (led by I. V. Prangishvili).

In addition to the scientific reports and discussions, participants at the conference visited the Kokhtla-Yarve Slate Chemical Combine and were familiarized with three automated systems for control of industrial processes:

1. The slate gas generator based on a UMI computer (automatic monitoring of processes and automatic control of the heat regime of the gas generators);

2. The process of ammonia production based on an M-6000 computer (monitoring production, automatic control of the work regime based on three-dimensional operation with human participation);

3. The process of carbamide production based on a YeS-1010 machine (monitoring production, automatic control of regulator assignments, optimization, and industrial reporting).

At the closing session the paths of future research in the field of control problems were outlined. The conference noted that the efforts of engineers and scientists should be directed to seeing that control problems in our country continue in the future to be resolved through a harmonious combination of work on fundamental problems of control theory and building new, highly efficient control hardware and progressive new technology with work to control complex organizational, economic production, and social systems.

The conference called on all specialists working in the control field to concentrate their efforts on fulfilling the tasks set down by the 26th CPSU Congress, bolstering the practical orientation of research, and doing everything possible to promote faster utilization of scientific results in the national economy and dissemination of results received on the broadest possible scale.

11,176
CSO: 1863/16

MODELING DEVELOPING SYSTEMS: WORK OF CONFERENCE REPORTED

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 142-145

[Article by V. V. Ivanov, A. A. Letichevskiy, and V. M. Yanenko]

[Text] The scientific councils on the problems "Cybernetics" and "Biological Physics" of the Ukrainian SSR Academy of Sciences, the L'vov department of the Institute of Biochemistry of the Ukrainian SSR Academy of Sciences, the Computer Center of the Institute of Applied Problems of Mechanics and Mathematics of the Ukrainian SSR Academy of Sciences, the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences, and the Institute of Physiology imeni A. A. Bogomolets of the Ukrainian SSR Academy of Sciences conducted the second consolidated conference of seminars on issues of modeling developing systems in the urban-type community of Slavskoye, L'vovskaya Oblast on 12-18 February 1981. The objectives of the conference were exchange of experience and raising the scientific level of the associates at large computer centers, scientific research institutes, and higher educational institutions. Participants at the conference presented reports and discussed timely topics at the round table. Sessions were held in four sections: general theory, numerical methods, applications in biology, and applications in economics.

The section "General Theory"

In the report "Fundamentals of Modeling Developing Systems" academician V. M. Glushkov, doctor of physicomathematical sciences V. V. Ivanov, and candidate of technical sciences V. M. Yanenko (Kiev, Institute of Mathematics of the Ukrainian SSR Academy of Sciences) reviewed the new generalizations and theoretical investigations of one class of dynamic models proposed in work [1]. The authors presented types of functions of the productivities of reproduction of developing systems for different statements of the problems. They propose evaluating the efficiency of the development of systems by the rate of change in their productivity and establish a number of patterns for the dynamics of system development.

Doctor of biological sciences Iu. G. Antomonov (Kiev, Institute of Cybernetics of the Ukrainian SSR Academy of Sciences) spoke in his report, "Modeling Developing Systems," about certain indicators (number, growth, change in

reaction), causes (environmental influence, adequacy of a system to its environment), and mechanisms of development (in ontogenesis and philogenesis, behavioral development). He named the complexity and level of organization of the system as the criterion of development. The speaker presented an evaluation of simple models of the course of development and discussed the simplicity and complexity of models of developing systems.

In his work "Investigation of the Behavior of Systems with Models of Probablistic Automata," candidate of physicomathematical sciences V. Ya. Valakh (Kiev, Institute of Cybernetics of the Ukrainian SSR Academy of Sciences) considered the theory of modeling probabilistic dynamic systems by means of automat models and presented mechanisms for formation of the so-called withdrawal "reflex" from "unpleasant" situations and drawing closer to "enticements."

The problems of describing the origin of biological information, the evolution of biosystems against a background of already existing information, the evolution of the individual, the optimality of development, and the existence of systems were treated in the report "Fundamental Problems of Theoretical Biophysics" by doctor of physicomathematical sciences D. S. Chernavsky (Moscow, Physics Institute imeni P. N. Lebedev of the USSR Academy of Sciences).

The report "Investigation of Stability and Oscillating Regimes in a New Class of Dynamic Models" by candidates of technical sciences V. M. Yanenko, Yu. P. Yatsenko, and O. V. Gorda (Kiev, Institute of Cybernetics of the Ukrainian SSR Academy of Sciences) reviewed the conditions of the current suboscillatory regimes in systems described within the framework of two, three, and five product dynamic models as proposed in [1-3]. In the case of singularly disturbed equations that describe a dynamic system, the possible behavior of oscillatory decisions of dynamic models was studied based on Tikhonov's theorem [4]. The report discussed the adequacy of the model and efficiency of its use to describe the ontogenesis of groundling embryos and to describe various types of interactions of lymphopoiesis and myelopoiesis in the organism.

The report "Spatial Self-Organization as the Result of Effectively Nonlocal Interactions in Dissipative Systems" by candidates of physicomathematical sciences B. N. Belintsev and M. A. Livshits and corresponding member of the USSR Academy of Sciences M. V. Vol'kenshteyn (Moscow, Institute of Molecular Biology of the USSR Academy of Sciences) presented an analytic study of the question of the mechanisms that provide the capability for spontaneous occurrence of spatial structures in unbalanced systems. It showed that relatively long-range negative cooperation alongside short-range positive cooperation are such mechanisms. Diffusion can play the role of the short-range mechanisms. By contrast, negative cooperation is associated with effectively nonlocal interactions.

Candidates of physicomathematical sciences M. A. Livshits, B. N. Belintsev, and G. T. Guriya, and corresponding member of the USSR Academy of Sciences M. V. Vol'kenshteyn (Moscow, Institute of Molecular Biology of the USSR Academy of Sciences), in their report "Spatial Differentiation in a Single-Component Dynamic System," identified two stages of differentiation: 1. the occurrence of the intitial spatial asymmetry, described as the appearance of the source

and confluence of the morphogene; (2) actual spatial differentiation in the cell ensemble. The authors proposed a reaction-diffusion model for the second stage and studied the nature of its stationary states, stability, and bifurcations.

In the report "Formal Description of the Evolution of Systems by Means of Structural-Functional Morphisms," candidate of physicomathematical sciences Ya. A. Dubrov, (L'vov, IPPMM [expansion unknown] of the Ukrainian SSR Academy of Sciences) proposed a formal definition of developing systems using the concept of structural-functional morphism. He showed that the given definition agrees with those available at the present time and demonstrated the possibility of describing the evolution of systems. The report outlined and investigated approaches to the study of the category of systems.

Candidate of technical sciences A. D. Krisilov (Odessa, OEE MO MGI [expansion unknown] of the Ukrainian SSR Academy of Sciences), in the report "Interaction of Developing Systems," reviewed the questions of formalizing the interaction of ecological and economic systems and the application of systems analysis to this problem. He proposed the use of an n-product dynamic model as a mathematical model of the systems "economy" and "natural environment" [2].

Roundtable Discussion — "General Theory"

The issues raised were the feasibility of "major" jumps in the evolution of dynamic systems and the development and investigation of optimal forms of dynamic models. The first set of questions was devoted to the evolutionary dynamics of change in certain properties of developing systems, in particular the need to accumulate excess "unneeded," but "not harmful" information for efficient system evolution (D. S. Chernavskiy), certain patterns of developing systems related to the dynamics of change in the reproductive productivity of developing systems (V. V. Ivanov), and some effective ways of utilizing energy in the abiotic stage of evolution (S. I. Kusen').

The Section "Numerical Methods"

The report by V. V. Ivanov and candidates of mathematical sciences P. N. Besarab and V. A. Lyudvinchenko (Kiev, Institute of Cybernetics of the Ukrainian SSR Academy of Sciences) "Package of Applied Programs for Numerical Realization of a Two-Product Model of Developing Systems" was devoted to the package, now under development, which consists of three groups of program modules: decision, nonstandard (with a right hand part) and system modules. The report analyzed the precision of some of the computing algorithms included in the package.

In their report "Numerical Solution of Certain Optimization Problems Based on a New Class of Dynamic Models," V. N. Denisyuk, V. M. Yanenko, and Yu. P. Yatsenko reviewed the problems of modeling competing interrelationships within the framework of a three-product model and identifying the parameters of a dynamic model.

B. N. Grabovskiy, P. N. Besarab, and candidate of physicomathematical sciences O. N. Odarich devoted their report, "Analytic and Numerical Solution of One Class of Mixed-Type Integral Equations," to the timely problem of modeling the processes of dispersion of streams of electrons in the nuclear structure of emulsions that occurs in the manufacture of microminiature models of printed plates and chips for highly productive computers.

P. A. Zhuk (L'vov, IPPMM Ukrainian SSR Academy of Sciences), in his report "The UNIMOD Package of Applied Programs and Possibilities for Using It To Model Complex Biological Systems" presented a method of constructing a package of programs that effectively fits the model of the object to the parameters of the original. The algorithm is based on an oriented graph whose apexes are blocks whose functions are dictated by the user.

The Roundtable Discussion "Numerical Methods"

The issues discussed were preserving the graphic quality and convenience of work with models when they are made more complicated, dividing technical, program, and personnel resources for modeling systems, and the development of modeling facilities as developing systems (V. V. Ivanov). A critical review of trends in the development of mathematical modeling in foreign countries was given (candidate of technical sciences Yu. A. Savostikskiy — Moscow, All-Union Scientific Research Institute of Systems Research, State Committee for Science and Technology of the USSR Council of Ministers) and a method was proposed for evaluating the efficiency of systems being introduced.

The Section "Applications in Biology"

Doctor of medical sciences L. S. Aleyev and candidate of technical sciences S. I. Garkusha made an analysis of the therapeutic process as a stochastic process in their report "Control of the Process of Patient Rehabilitation in Neurology." They proposed an adaptive rule for making a decision that is one of the modifications of a composite decision.

The report of candidate of biological sciences T. M. Yeroshenko (Moscow, Moscow State University imeni M. V. Lomonosov) "Characteristics of the System of Body Weight Regulation in Growing Animals" was devoted to the conception of "setting animal weight." The report took note of approaches that consider the relationship between reproduction of the live weight of the organism and the total weight of the body, evaluted the role of growth hormones, and proposed a corresponding mathematical model.

In their report "Modeling a Developing Neuron Structure," candidates of biological sciences I. A. Rybak and V. N. Yefimov (Rostov-na-Donu, NIINK [possibly Scientific Research Institute of Neurocybernetics]) showed the interaction of the neuron structure with the environment. In the process of interaction with the environment the system, by rearrangements of the structure of linkages, arrives at a type of reaction that insures receiving the essential (maximum) amount of substrate.

The report by A. I Bogatyr' (Kiev, NII NKh [possibly Scientific Research Institute of Neurosurgery] of the Ukrainian SSR Ministry of Health), "Rhythms of Homeokinetic Regulation and Adaption," was devoted to a quantitative evaluation of the relationship between biorhythms (period of eight hours) and moments when organism adaptation is disrupted. This is very important for planning operations in neurosurgery.

The report "Theoretical Aspects of Adaptation" by M. Yu. Antomonov (Kiev, KNII GKKh [expansion unknown] of the Ukrainian SSR Ministry of Health) gave a critical analysis of approaches to defining the concept of adaptation in living systems. The report reviewed an existing system for automating scientific research that makes it possible to evaluate the degree of organism adaptation to harmful environmental factors.

V. V. Ivanov, candidate of biological sciences N. I. Lysenko, and V. M. Yanenko (Kiev, Institute of Cybernetics of the Ukrainian SSR of the Academy of Sciences and IMBG [possibly Institute of Molecular Biology and Genetics] of the Ukrainian SSR Academy of Sciences), in the report "Application of a New Class of Dynamic Models in Genetics and Plant Breeding (with the Example of Sugar Beets)," considered five variations of models of the process of sugar accumulation in inbred sugar beet lines. Averaged estimates of the parameters of yield and sugar content were used for modeling and preliminary recommendations were presented for predicting optimal hybrid vigor in sugar beets. The report "Identifying the Latent Characteristics of Developing Systems by Spectral Analysis" by D. I. Sanagurskiy, R. I. Gnat'yev, candidate of biological sciences Ye. A. Goyda, and T. I. Murskaya (L'vov, LO IBKh imeni A. V. Palladin [expansion unknown] of the Ukrainian SSR Academy of Sciences) gave an assessment of the correlation between parameters (rate of oxygen consumption, transmembrane potential, and overoxidation of the lipids of the groundling embryo) with different stages of its morphogenesis and the role of calcium ions in the process of fertilization and development of the embryo.

In their report "Complex of Initial Biophysical Parameters for Modeling the Regulatory Systems of Early Development of Animals," Ye. A. Goyda, I. O. Mukalov, D. I. Sanagurskiy, and N. S. Stel'makh (L'vov, LO IBKh imeni A. V. Palladin of the Ukrainian SSR Academy of Sciences) gave an evaluation of the process of registration of different biophysical parameters of groundling embryos in real time by means of computer and assessed the influence of colchicine in the mitotic cycle.

The report of candidate of physicomathematical sciences V. G. Levadnyy (Moscow, Scientific Council on Cybernetics of the USSR Academy of Sciences), "Optimal Functional Parameters of Respiration," was devoted to a consideration of different optimization problems, among which the problem of minimum full energy expenditures in the regime of stationary functioning of the external respiratory system of the organism is adequate.

Candidates of physicomathematical sciences A. L. Asachenkov, L. N. Belykh, S. M. Zuyev, and A. A. Romanyukha (Novosibirsk, Computer Center of the Siberian Department of the USSR Academy of Sciences) presented a report entitled

"Mathematical Modeling of Infectious Diseases and Methods of Processing Clinical and Laboratory Data." The report was based on a base model proposed earlier by academician G. I. Marchuk [6]. One of the interesting results that has been applied in the clinic is the therapeutic tactic of aggravating the condition of the patient.

The report "Regulation of the Humoral Immune Response" by candidate of biological sciences R. N. Stepanenko (Moscow, 2nd Moscow State Order of Lenin Medical Institute imeni N. I. Pirogov of the RSFSR Ministry of Health) discussed the principal types of intercellular interactions of the immune system. The speaker proposed a block diagram of the immune system and identified the "channels" that regulate the humoral immune response.

The report "Theory of the Immune Network" by candidate of biological sciences V. G. Nesterenko (Moscow, Institute of Epidemiology and Microbiology imeni N. F. Gamaleya of the USSR Academy of Medical Sciences) reviewed the progressive formation of the theoretical conception of the immune network. Within this framework it is possible to explain a large number of factors, and in particular those related to high- dose and low-dose immunological tolerance. The report presented original results on the "fine" immunological mechanism for regulating the differentiation of target cells, which leads to further elaboration of the theory [7].

The report entitled "Development of Mathematical and Experimental Models of Immunological Tolerance and the Mechanisms of Regulation of Adaptive Differentiation Thymocytes and Endocrine Regulation of the Immune Response" by doctors of biological sciences L. N. Fontalin and V. G. Nesterenko, doctor of medical sciences B. B. Fuks, candidate of biological sciences G. M. Zhuravel', doctor of medical sciences E. V. Gyulling, academician V. M. Glushkov, doctor of physicomathematical sciences V. V. Ivanov, candidate of technical sciences V. N. Yanenko, and S. N. Garkusha emphasized the important role of regulation of the redistribution of immune system resources to support the "internal" function and to perform external (killer, suppressor, and other) functions. The report reviewed a further elaboration of V. N. Glushkov's mathematical model [1] for certain problems of modeling immunological tolerance and the like, introduced new experimental models for the purpose of finding new regulatory elements of the immune system with conditions for switching the system from the immune response to immunological tolerance, and presented findings from the literature that confirm the adequacy of the model.

In his report entitled "Characteristics of Modeling the Blood Formation System," candidate of biological sciences A. Ya. Monichev (Gor'kiy, Scientific Research Institute of Mathematics at Gor'kiy State University) presented a new variation of the mathematical model of regulation of hemopoiesis [8], proposed possible mechanisms of regulation, and demonstrated the adequacy of the model.

Candidate of technical sciences V. A. Geodokyan (Institute of Developmental Biology imeni N. K. Kol'tsov of the USSR Academy of Sciences) gave a report entitled "Living Systems and the Conception of Information." The report was devoted to the search for conceptual models and evaluated the difference between

ordering and organization and the role of information in living systems, noting that the relationship between the zygote and the organism is such that the mature organism, as the result of information work, possesses a minimum of information potential and a maximum volume of information.

The Roundtable Discussion "Conception of Information"

The problems of describing the origin of biological information were discussed (D. S. Chernavskiy) and an evaluation was given of different approaches to defining the probability of random occurrence of a biocode [9]), the latest results on the hypercycle, and related probablistic assessments of the occurrence of the translation system [10]. Differentiation models were considered which make it possible to represent the selection of one biocode owing to the instability of the states of the system and the accumulation of "ballast" information. In such models, however, threshold cycles and foreign attractors are also possible (doctor of physiocomathematical sciences A. N. Molchanov, Scientific Institute of the Computer Center of the USSR Academy of Sciences) [11], which diminishes the probabilistic evaluations of the origin of the biocode. The evolutionary mechanisms of differentiation of the sexes were considered and the causes of sexual dimorphism under extreme environmental conditions were clarified.

The Section "Modeling Ecological and Economic Systems and Scientific-Technical Progress"

In his report "The Evolution of the Concept of the Stationary Regime," A. N. Molchanov showed how the theory developed from the concept of one stationary stable state toward the concepts of stationary states. The author presented an analysis of different types of oscillatory behaviors of biosystems in which such states as quasistochasticity are possible and demonstrated the applicability of these results to the problems of mathematical ecology and population genetics [11].

Candidate of philosophical sciences V. Ye. Khmel'ko (Kiev, Institute of Party History of the Central Committee of the Ukrainian Communist Party, Branch of the Institute of Marxism-Leninism of the CPSU Central Committee) devoted his report, "The Sociological Approach to Development of Models of the Development of the Macrostructure of Public Production," to an analysis of the tendency in development of reproduction evaluation of the change in the dominant factors in social production of life, and consideration of its character. The report reviewed transitory change in the proportion of expenditures of social labor in the primary spheres of reproduction of life itself (means of existence and human beings themselves). A trend was identified for the proportion of expenditures of labor for information production to become paramount, and it is expected in the future that the reproduction of human beings themselves will be the dominant factor.

Criticism of several bourgeois conceptions of the development of public production was given.

F. A. Zhuk's report "One Approach to Formalization of Purposeful Systems and Determining Conditions for Their Progressive Development" presented a general theoretical consideration of systems of various types of interaction in purposeful systems which are formed with due regard for a set of permissible environmental strategies.

Candidates of technical sciences A. A. Yakunin and Yu. A. Yershov (Dnepropetrovsk, Main Information and Computer Center of the Ukrainian SSR Ministry of Ferrous Metallurgy) presented a report entitled "Integrated Automated Control Systems of the Ukrainian SSR Ministry of Ferrous Metallurgy," dealing with the questions of setting up and operating the information base of an automated control system and evaluating the efficiency of several elements of production. The report "Structural Modeling of Socioeconomic Development of a Region" by doctor of economic sciences V. A. Podsolenko (Dnepropetrovsk, IEP [possibly Institute of Economic Planning] of the Ukrainian SSR Academy of Sciences) considered numerous issues: methodological questions of setting up systems for modeling regions, the structure of mathematical and information software, and the possibility of solving optimization problems involving optimal distribution of a region's resources used in industry and to satisfy social needs. V. V. Ivanov, V. M. Yanenko, and U. Ye. Galiyev (Kiev, Institute of Cybernetics of the Ukrainian SSR Academy of Sciences, Alma-Ata, AGU [possibly Alma-Ata State University]) presented a report entitled "Modeling Scientific-Technical Progress on the Basis of Dynamic Models According to V. M. Glushkov." In the report they proposed evaluating the efficiency of development of systems by the rate of change in productivity. They demonstrated several patterns of systems in development and give examples of evaluating the efficiency of development of scientific-technical progress.

Summarizing the above, we feel that the consolidated conference of seminars on questions of modeling developing systems was mutually beneficial, that the reports aroused great interest and participants at the conference were rewarded with new, specialized knowledge.

FOOTNOTES

1. V. M. Glushkov, "One Class of Dynamic Macroeconomic Models," UPRAVLYAYUSCHIYE SISTEMY I MASHINY, 1977, No 2, pp 3-6.

2. V. M. Glushkov, V. V. Ivanov, and V. M. Yanenko, "A New Class of Dynamic Models and Its Application in Biology. I, II," KIBERNETIKA, 1979, No 4, pp 131-139; 1980, No 4, pp 109-117.

3. V. M. Glushkov, V. V. Ivanov, and Yu. P. Yatsenko, "Analytic Investigation of One Class of Dynamic Models. I, II," KIBERNETIKA, 1980, No 2, pp 1-12.

4. A. B. Vasil'yeva, and V. F. Butuzov, "Asimptoticheskoye Razlozheniye Resheniye Singulyarno Vozmushchennykh Uravneniy" [Asymptotic Expansion of Singularly Disturbed Equations], Moscow, "Nauka", 1974, 272 pages.

5. V. M. Glushkov, V. V. Ivanov, V. M. Yanenko, Yu. P. Yatsenko, S. V. Kakichev, "Means of Modeling Developing Systems," Preprint of the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences, No 80-37, Kiev, 1980, 58 pages.

6. G. I. Narchuk, "Matematicheskiye Modeli v Immunologii" [Mathematical Models in Immunology], Moscow, "Nauka", 1980, 264 pages.

7. V. G. Nesterenko, "Immunological Regulation of the Differentiation of Somatic Cells," USPEKHI SOVREMENNOY BIOLOGII 1980, 90, Vyp 2, pp 211-220.

8. V. P. Gruzdev, A. Ya. Monichev, and Ye. N. Shcherbova, "Issues of Mathematical Modeling of the Blood Formation Process (Truncal Cell)" VOPR. KIBERNETIKI 1979, Vyp 49, pp 64-106.

9. ZHURNAL VSESOYUZNOGO KHIMICHESKOGO OBSHCHESTVA IM. D. I. MENDELEYEVA 1980, No 3, 152 pages; No 4, 152 pages.

10. M. Eigen, and P. Schuster, "The Hypercycle," 1-3, NATURWISSENSCHAFTEN 1977, 64, No 11, pp 54-572; 1978, 65, No 1, pp 7-37; 1968, 62, No 7, pp 341-375.

11. A. M. Molchanov, and A. D. Bazykin (editors), "Faktory Raznoobraziya v Matematicheskoy Ekologii i Populyatsionnoy Genetike" [Diversity Factors in Mathematical Ecology and Population Genetics], Pushchino, Nauchno-Isseldovatel'skiy Vychislitel'nyy Tsentr SSSR, 1980, 196 pages.

11,176
CSO: 1863/16

4TH ALL-UNION SEMINAR 'PARALLEL PROGRAMMING AND HIGH-PRODUCTIVITY STRUCTURES'

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 145-146

[Article by G. Ye. Tseytlin and L. I. Nagornaya]

[Text] Structural Programming in Systems of Algorithmic Algebras

A seminar entitled "Parallel Programming and High-Productivity Structures" was held in Kiev on 25-26 February 1981. The seminar was devoted to problems of structural programming in systems of algorithmic algebras and was organized by the section "Homogeneous Computer Systems, Media, and Distributed Systems" of the Scientific Council on Computer Technology and Control Systems of the USSR State Committee for Science and Technology, the section "Computer Technology" of the All-Union Scientific-Technical Society of Radio Engineering and Telecommunications imeni A. S. Popov, and the Scientific Council on the Problem "Cybernetics" of the Ukrainian SSR Academy of Sciences. Seventy-three specialists in programming and computer technology from various organizations in Kiev, Moscow, L'vov, Dnepropetrovsk, Kazan', Novosibirsk, and other science centers of the country took part in the work of the seminar.

The program of the seminar included a series of plenary survey reports devoted to the development of important areas of programming technology and treating the general problem area of structural programming.

The report "The Organization of Computations on Multiprocessor Computing Systems" by V. M. Glushkov, Yu. V. Kapitonova, and A. A. Letichevskiy reviewed the questions of the development of hardware and software for multiprocessor computing systems based on the macroconveyor principle. The report formulated the principal results of the algebraic theory of structures of data oriented to synchronous and asynchronous multiple processing. Significant attention was devoted to the method of formalized technical specifications, which is oriented to the development of parallel programs and to the family of parallel programming languages being developed for RVM [expansion unknown]. The expressive capabilities of the proposed linguistic means were illustrated with examples of computer mathematics problems.

In his report entitled "Structural Programming and P-Technology," I. V. Vel'bitskiy described the present state and prospects for development of

P-technology, a widespread method of designing programs and discussed the conceptual foundation of the technology. The report gave special attention to the graph form of describing data in combination with analytic representation of programs in P-metalanguage on standard memory structures using the apparatus of semantic procedures. A number of practical results obtained from the application of P-technology were described and some problems for further research were formulated.

V. N. Red'ko's report "Universal Program Logic" was devoted to an analysis of investigation based on program logics which have emerged and are developing at the boundary between mathematical logic and programming theory. Special attention was devoted to universal program logic, based on the composition approach. In the program logic constructed the conclusions adequately reflect the process of program design. The concepts of semantic structure and semantic program design are refined on this basis.

The report "Multilevel Structural Program Design" by G. Ye. Tseytlin proposed a method of program development that combines the apparatus of systems of algorithmic algebras with the theory of formal language models. This method makes it possible to formalize the structure of the program at each level of design and also permits switching by levels in the process of detailing it. The report gave basic results on the schematology of structural parallel programming based on the apparatus of modified systems of algorithmic algebras and the theory of grammatical and automat language models. The method was applied in developing the instruments of structural parallel programming, one system of parallel translation, certain components of the software of homogeneous computing systems, minicomputers, and microcomputers. The report formulated certain open questions within the framework of the further development of this area of research.

In addition to the plenary sessions there were three topical sessions at which the following communications were presented.

The Session "Instruments of Structural Programming in Systems of Algorithmic Algebras"

- V. P. Zhidakov, "Structural Design of Complex Problems";
- V. P. Gritsay and O. I. Tal'yanskaya, "Realization of a System That Translates from the Language of Logical Systems of Algebraic Algebras into Pl-1";
- Yu. A. Yushchenko, "Analytic Transformations of Regular Program Circuits in S-Algebras on an L2B Base";
- M. S. Burgin, "Register Transformation, Multidimensional Statements, and Their Compositions";
- V. P. Panfilenko, "One Effective Algorithm for Testing the Correstness of Parallel Regular Program Circuits."

The Session "Processes of Symbol Processing Systems of Algorithmic Algebras"

- L. I. Nagornaya, "Structural Description of the PARTRAN System, a Multilayer Parallel Conveyor-Type Translator";
- Ye. P. Batrak, "Multilevel Structured COBOL, the Input Language of the PARTRAN System";
- M. Ye. Berkovich and M. P. Milov, "Realization of Parallelism Facilities in the ADAM Language";
- M. N. Muchnik, "Structural Description of the Software Components of Minicomputers and Microcomputers";
- V. P. Zhidakov, T. D. Lomidze, and A. A. Vasil'kov, "Information Software of a Structured Data Bank";
- A. V. Babichev, "Non-Sorting Analysis of KS-Languages."

The Session "Structural Designing of System Control Processes"

- N. S. Maksimov, "Algorithmic Models of the Functioning of the Display Complexes of YeS [Unified System] Computers";
- A. V. Anisimov, Yu. Ye. Boreysha, and M. V. Nikolayev, "System for Modeling Multiprocessor Systems";
- V. K. Vodop'yanov, "Modification of Linguistic Means Oriented to Describing Industrial Processes";
- M. Ye. Berkovich and I. B. Osipov, "Solving Bookkeeping Problems in a Locally Distributed Computer System";
- T. K. Terz'yan, "Linguistic and Program Software of Systems To Automate the Designing of Industrial Processing."

The work of the seminar concluded with a discussion entitled "The Role of Formalization in Program Design," which promoted a lively exchange of opinions and experience in the basic areas of the technology and methodology of structure parallel programming and further stimulated work to apply the formal apparatus of algorithmic algebras.

11,176
CSO: 1863/16

3RD SEMINAR 'INTERACTIVE SYSTEMS'

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 pp 146-147

[Article by L. N. Nekrasova and M. G. Khoshtaria]

[Text] The 3rd seminar was held on 15-21 March 1981 in the Georgian city of Borzhomi.

The seminar was organized by the Georgian republic seminar "Interactive Systems" of the Council of Young Scientists of the Central Committee of the Georgian Komsomol, the Council on Automation of Scientific Research of the Presidium of the USSR Academy of Sciences, the Institute of Cybernetics of the Georgian SSR Academy of Sciences, the Computer Center of the Georgian SSR Academy of Sciences, the Georgian Scientific Research Institute of Scientific-Technical Information, the Information and Computing Center of the Georgian SSR Ministry of Finance, and the Institute of Cybernetics of the Ukrainian SSR Academy of Sciences.

The seminar drew the participation of 256 scientists from 28 cities of the Soviet Union. More than 100 science centers and organizations of the USSR Academy of Sciences, the academies of sciences of the Union republics, the USSR State Committee for Science and Technology, the State Committee on the Use of Atomic Energy, and various ministries and departments (including organizations of the ministries of education of the USSR and the Union republics) were represented.

Twenty-one reports were presented in two plenary sessions and eight lecture periods. V. V. Chavchanidze, chairman of the organizing committee and academician of the Georgian SSR Academy of Sciences, gave an introductory talk. Discussions were held in two sections on the basis of abstracts of 130 reports that were submitted and published in advance.

Section 1. Problem-Oriented Interactive Systems

This section was led by professors A. S. Vayradyan and V. I. Varshavskiy; L. N. Nekrasova was the learned secretary.

The issues considered were:

1. Problem-oriented interactive computer complexes. Interactive packages of applied programs and organizations;

2. Interactive systems based on microcomputers;

3. Interactive machine graphics.

Section 2. Interactive (Dialogue) Languages

The leaders of this section were professor A. V. Gladkiy and doctor of philological sciences and professor E. S. Skorokhod'ko; the learned secretary was candidate of physicomathematical sciences M. G. Pkhovelishvili.

The issues considered were:

1. Question-answer systems for communication with data bases and natural languages;

2. Interactive programming languages;

3. Linguistic problems of diaglogue (interaction).

A broad range of problems related to the development, realization, and application of interactive systems was considered at the seminar. Great attention was devoted to improving the forms and languages for communication between the user and the computer (the report "Interactive Programming Languages" by S. P. Prokhorov, the report "Linguistic Means of Interaction in ASAS [expansion unknown]" by Yu. M. Borodyanskiy, L. N. Nekrasova, I. K. Psikunova, and others); methodological problems of constructing problem-oriented interactive systems (the report "Methods of Generating Interactive Routed Systems" by Ye. R. Yushchenko, V. S. Sadovenko, et al., "Principles of Construction of Industrial Systems That Provide Dialogue with the User in a Limited Natural Language" by E. V. Popov, and "Automated Design of the Structure of a Multiprocessor Computing System in the Interactive Mode" by A. S. Vaydaryan); P-technology (the report "Current State of P-Technology and Means of Supporting It in YeS [Unified System] Computers, SM [Small Computer] Machines, and the BESM-6" by I. V. Vel'bitskiy); questions of their mathematical, program, linguistic, and informational software (the report "The Linguistic Approach to the Construction of Computer Complexes To Process Images of the Geometric Drawing Type" by A. L. Sinyavskiy, "The Ideas of M. N. Bakhtin on Expression and Dialogue and Their Importance for the Formal Semantics of Natural Language" by A. V. Gladkiy, "Presumptions and Their Role in Human Interaction with the Machine" by Ye. V. Paducheva, "Linguistic Software in Contemporary Text Information Processing Systems" by Yu. D. Apresyan, the "DESTA Interactive Natural Language System" by V. A. Levitskiy, and "Compiling and Managing Vocabularies in the Interactive Mode" by E. F. Skorokhod'ko.)

Work on the use of minicomputers (the report by A. S. Barinov) and interactive graphic systems (the report "Basic Software of Machine Graphics for Automating Scientific Research and Design in Construction Using SM Computers" by A. A. Lyashchenko) was broadly represented.

In addition to the above-mentioned reports, systems that have been realized in practice were presented at the seminar (I. V. Sergiyenko, A. S. Stukalo, et al., "The ASPP FAP Interactive Automated System for Program Preparation on YeS Computers"), and the report "Modeling the Transportation Situation in the Interactive Mode on the INTRANS-2 System" by V. I. Gritsenko and V. M. Miroshnichenko).

The following reports were presented at the plenary session: "Some Questions of Problem-Oriented DISP [possibly Interactive Planning Information Systems]" by V. V. Aleksandrov, "Timely Questions of Dialogue" by V. E. Vol'fangangen, "Model of Protocol for Exchange in Interactive Systems" by V. I. Varshavskiy, "Methodology for Structured Design of Interactive Programs" by S. V. Yastzhembskiy, and "The Ideas of M. N. Bakhtin on Expression and Dialogue and Their Importance for the Formal Semantics of Natural Language" by A. V. Gladkiy.

Within the framework of the overall seminar a special seminar entitled "The Use of Minicomputers and Microcomputers in Interactive Systems in the Automation of Scientific Research" was held under the direction of professor I. I. Malashinin, chairman of the Commission on Interactive Systems of the Council on Automation of Scientific Research of the Presidium of the USSR Academy of Sciences.

During the work of the seminar scientists exchanged opinions on timely problems of human interaction with computers in solving various classes of problems. The most interesting work and lines of activity were noted, scientific contacts were made among many scientific collectives, and the paths of future development of interactive systems were outlined.

It can be stated that in the current phase of development of computer technology and methodological, linguistic, mathematical, program, and information software with a growing role of automated systems and their widespread introduction in many fields of science and the economy, the question of the human relationship to computing systems and human interaction with the computer is becoming critical. Contemporary interactive systems allow human interaction with the computer in professional languages, limited natural languages, or languages that are close to natural languages. They greatly broaden the sphere of application of computers, the classes of problems that can be solved, and the range of users.

The seminar, which reviewed the state of research, development, and accomplishments with respect to interactive systems, highlights the specialization of many science centers in particular fields of designing interactive systems and the current level of automated interactive systems being set up.

The resolution of the seminar should play a positive role in coordinating scientific organizations working on the development of interactive systems and developing methods to standardize interactive systems in order to raise the efficiency of use of computer technology in many spheres of scientific research and the national economy.

It was recommended that the 4th All-Union Seminar be held in the first half of 1982.

11,176
CSO: 1863/16

INTERNATIONAL EXHIBITION 'COMMUNICATIONS-81' IN MOSCOW

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 8, Aug 81 p 38

[Article by Yu. M. Samoylov, senior editor of Soviet section of international exhibition "Communications-81," in the section "International and Foreign Exhibitions"]

[Text] This year, our country for the second time (after 1975) will be the site of an international review of communications systems and equipment. The firms and organizations of more than 20 of the world's countries will demonstrate models of their products at the international specialized exhibition "Communications-81" [Svyaz'-81] that will be held from 2 through 16 September in Moscow's Sokolniki Park.

The USSR has one of the most important expositions. It has more than 3,000 exhibits providing full representation on the achievements and prospects for development of communications equipment, beginning with the oldest, mail, and ending with space. The exposition will have eight thematic sections: "Satellite Communications," "Radio Communications," "Television and Radio Broadcasting," "Mail," "Radio Electronic Components and Materials for Communications Equipment," and others.

Products from the enterprises of the Ministry of Instrument Making, Automation Equipment and Control Systems will be widely shown at the exhibition. Thus, specialists and visitors will see models of data transmission equipment in the "Terminal Equipment, Channels and Networks for Communications." The Nal'chik Telemechanical Equipment Plant imeni the 50th Anniversary of the USSR, for example, will demonstrate signal converters and a subscriber station. The UPS-50/200 signal converter that will be on display is designed for synchronous or asynchronous serial data transmission over unswitched telegraph networks with four- or three-wire termination (one wire common) and can be used in various automated control systems. The unit provides for duplex or half-duplex data exchange at tranmission rates of 50, 75, 100 or 200 bits/sec in the asynchronous mode or up to 200 bits/sec in the synchronous mode.

The "Modem-1200KN" signal converter is used for synchronous or asynchronous serial data transmission over voice-frequency telephone channels and physical lines with two- or four-wire termination. It comes in six versions and can be used in various automated control systems. The unit transmits data over a private line at 1200 bits/sec in the asynchronous mode, 600 or 1200 bits/sec in the synchronous, and up to 75 bits/sec over the return channel.

The UVTK-KI subscriber station is used to build subsystems for acquisition and distribution of data in automated control systems. Through modems and terminals connected to the subscriber station, a number of operators can remotely access a data processing center computer. Built with the "Elektronika-60" computer, the UVTK-KI station provides for data tranmission from remote (to 2 km) teletypes, computers located nearby and data transmission equipment; it operates over telephone, switched and dedicated, city and intercity, as well as physical lines at 600 and 1200 bits/sec, and over telegraph dedicated lines at 50, 75, 100 or 200 bits/sec.

The L'vov "Mikropribor" Production Association imeni the 60th Anniversary of the Soviet Ukraine will show in this section the MD-4801 analog storage unit. It is used in data processing and transmission units in large analog-digital measuring systems as well as metrological equipment and systems. The unit features high speed and low error rate for entry and storage of the momentary value of an analog signal.

L'vov products will also be demonstrated in the "Radio Measuring Equipment" division. The Shch-48000 general-purpose digital voltmeter, in particular, is on the standard of the world's best. It is designed to meter voltages of direct and alternating current, strength of direct current and resistance. It is used both under laboratory conditions (for checking and repairing electronic apparatus) and in monitoring and control systems (for converting transducer output signals into digital code). The instrument is highly sensitive, considerably fast and highly immune to noise and has a low error rate in metering.

The F-4834 digital voltmeter for direct current meters voltage in the range from 5 microvolts to 1000 V and is used in a four-channel biochemical automatic analyzer as the meter unit in automatic monitoring and control systems; it is also used as an independent instrument in laboratories and industrial enterprises.

The F-4801 digital voltmeter for direct current meters voltage and the relationship between two voltages of direct current. It is used as an autonomous instrument or combined with meter converters to meter various passive and active parameters of electrical circuits. The direct current voltage meter range is 0.1, 1, 10 and 100 V.

The F-4852 digital combined instrument meters the root-mean-square value of alternating voltage and current, direct voltage and current and active power in single-phase circuits. It is used in information and metering systems and systems for automatic monitoring of signals of alternating and direct voltage and current. Compared to similar foreign and domestic models, it features high speed, precision and general-purpose application.

Models of products from the Zhitomir"Elektroizmeritel'" Production Association will be demonstrated on the stands of the Soviet exposition. Among them are electric meter combined instruments, signal and spectrum analyzers and a millitesla-meter. Compared to similar foreign models, the F-4372 with digital and CRT display features higher metering accuracy and the capability of metering currents plus an expanded frequency range. It meters strength and voltage of direct and alternating current, direct current resistance and frequency of electrical signals, and is used

to observe and meter parameters of signals for a CRT screen, analyze amplitude-frequency characteristics of resonance systems of electronic devices and to analyze audio frequency devices.

Two other Minpribor enterprises will be represented at the exhibition. The Special Process Design Bureau of Process Control Computers (Orel) will invite specialist attention to its development, an autonomous information recorder; the Omsk "Elektrotochpribor" Plant will display the F-283 digital voltmeter for metering voltage in direct current circuits. Compared to similar models, the instrument features high sensitivity, noise immunity, reliability and broad functional capabilities.

8545
CSO: 1863/256

PUBLICATIONS

TABLE OF CONTENTS OF JOURNAL 'KIBERNETIKA', JULY-AUGUST 1981

Kiev KIBERNETIKA in Russian No 4, Jul-Aug 81 p 149

[Table of Contents]

TABLE OF CONTENTS FROM JOURNAL 'ELECTRONIC MODELING', SEPTEMBER-OCTOBER 1981

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 81 pp 1-2

[Text] Table of Contents Page

8545
CSO: 1863/10

PROBLEMS OF CYBERNETICS: CONTROL OF SYSTEMS DEVELOPMENT

Moscow VOPROSY KIBERNETIKI: UPRAVLENIYE RAZVITIYEM SISTEM in Russian 1979 pp 2-3

[Annotation and table of contents from the collection "Problems of Cybernetics: Control of Systems Development", Nauchnyy sovet po kompleksnoy probleme "Kibernetika" AN SSSR]

[Text] Different approaches to problems of describing developing systems in the medium and formalization of the relationships of the determining properties of a system such as reliability, orderliness, efficiency and optimality are studied. The possibilities of organizing character structures and natural language structures and the possibilities of reducing them are studied. Different aspects of systems development are considered. Applications of development theory to problems of creating computer networks that control decision-making processes, coordination and monitoring with respect to problems of automated design systems development are given.

Collection VK-56 is issued as information materials of the Scientific Council on the Complex Problem "Cybernetics," USSR Academy of Sciences.

Contents

6521
CSO: 1863/5

- END -